

BEST AVAILABLE COPY

(19)日本国特許庁 (JP)

(12) 公開特許公報 (A)

(11)特許出願公開番号
特開2000-23190
(P2000-23190A)

(43)公開日 平成12年1月21日 (2000.1.21)

(51)Int.Cl. ⁷	識別記号	F I	マーク7(参考)
H 0 4 N 11/04		H 0 4 N 11/04	B 5 C 0 5 7
7/32		H 0 3 M 7/36	5 C 0 5 9
// H 0 3 M 7/36		H 0 4 N 7/137	Z 5 J 0 6 4

審査請求 未請求 請求項の数25 FD (全46頁) 5-3

(21)出願番号 特願平10-196769
(22)出願日 平成10年6月26日(1998.6.26)

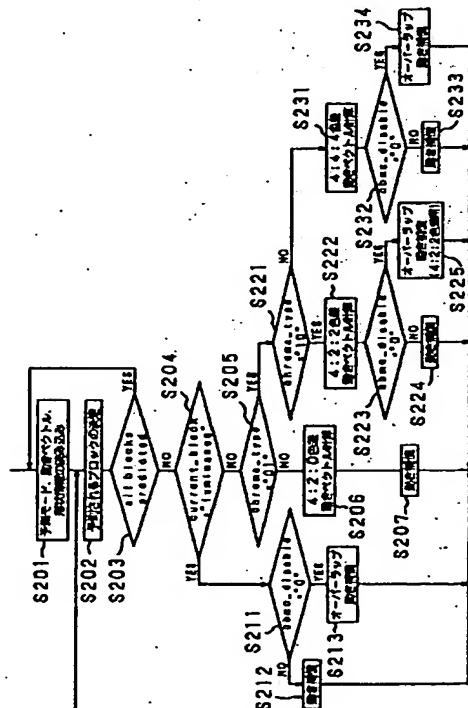
(71)出願人 000002185
ソニーリミテッド
東京都品川区北品川6丁目7番35号
(72)発明者 春原 修
東京都品川区北品川6丁目7番35号 ソニーリミテッド
(72)発明者 矢ヶ崎 陽一
東京都品川区北品川6丁目7番35号 ソニーリミテッド
(74)代理人 100067736
弁理士 小池 晃 (外2名)

最終頁に続く

(54)【発明の名称】 動きベクトル生成方法、画像符号化装置、動き補償方法、動き補償装置、及び提供媒体

(57)【要約】

【課題】 4:2:2、4:4:4フォーマットの色差ブロックの動きベクトル、動き補償を有効に行わせる。
【解決手段】ステップS201でマクロブロックの予測モード、動きベクトル、形状情報を読み込み、ステップS205、S221で、色差フォーマットが4:2:0、4:2:2、4:4:4のいずれかを判別し、それぞれの色差フォーマット毎に動きベクトル計算方法を選択している。また、ステップS223、S232で、オーバーラップ動き補償を行うか否かをフラグ obmc_disable で判別し、オーバーラップ動き補償と、通常の動き補償とを切換選択している。



【特許請求の範囲】

【請求項1】 画像の輝度成分と色差成分とをそれぞれブロック化して各ブロック毎の動きベクトルを生成する動きベクトル生成方法において、上記輝度成分のブロックに対して得られた動きベクトルに基づいて、対応する色差成分のブロックの動きベクトルを生成することを特徴とする動きベクトル生成方法。

【請求項2】 上記色差成分として、複数種類の色差フォーマットのいずれかが選択された色差成分を用い、この色差フォーマットの種類に応じて、上記輝度成分のブロックに対して得られた動きベクトルに基づく色差成分のブロックの動きベクトルの計算を行うことを特徴とする請求項1記載の動きベクトル生成方法。

【請求項3】 上記複数種類の色差フォーマットは、
4:2:0フォーマット、4:2:2フォーマット、
4:4:4フォーマットであることを特徴とする請求項2記載の動きベクトル生成方法。

【請求項4】 上記動きベクトルの計算を、上記色差フォーマットの種類と、マクロブロックの予測モードとに応じて切換選択することを特徴とする請求項2記載の動きベクトル生成方法。

【請求項5】 画像の上記輝度成分及び色差成分と共に、画像の形状情報が供給され、この形状情報に応じて、上記色差成分のブロックの動きベクトルを計算することを特徴とする請求項2記載の動きベクトル生成方法。

【請求項6】 上記動きベクトルは、ブロック毎に予測符号化を行うための動き補償に用いられるものであることを特徴とする請求項2記載の動きベクトル生成方法。

【請求項7】 画像の輝度成分と色差成分とをそれぞれブロック化して各ブロック毎の動きベクトルを求め、動き補償を伴う予測符号化を行う画像符号化装置において、

上記色差成分として、複数種類の色差フォーマットのいずれかが選択された色差成分を用い、上記輝度成分のブロックの動きベクトルを検出すると共に、得られた輝度成分の動きベクトルに基づき、上記色差成分の色差フォーマットの種類に応じて選択された計算により、対応する色差成分のブロックの動きベクトルを生成する動きベクトル検出手段と、

この動きベクトル検出手段により得られた動きベクトルを用いて動き補償を行う動き補償手段と、この動き補償手段からの出力と入力画像とに基づいて予測符号化を行う符号化部とを有することを特徴とする画像符号化装置。

【請求項8】 上記複数種類の色差フォーマットは、
4:2:0フォーマット、4:2:2フォーマット、
4:4:4フォーマットであることを特徴とする請求項7記載の画像符号化装置。

【請求項9】 上記動きベクトルの計算を、上記色差フ

50 オーマットの種類と、マクロブロックの予測モードとに応じて切換選択することを特徴とする請求項7記載の画像符号化装置。

【請求項10】 画像の上記輝度成分及び色差成分と共に、画像の形状情報が供給され、この形状情報に応じて、上記色差成分のブロックの動きベクトルを計算することを特徴とする請求項7記載の画像符号化装置。

【請求項11】 画像の輝度成分と色差成分とをそれぞれブロック化して各ブロック毎に求められた動きベクトルに基づいて動き補償を行う動き補償方法において、

上記色差成分のブロックの動きベクトルは、上記輝度成分のブロックに対して検出された動きベクトルに基づいて生成されたものであり、

上記色差成分のブロックの動きベクトルに応じて、当該色差成分のブロックの動き補償を行うことを特徴とする動き補償方法。

【請求項12】 上記色差成分として、複数種類の色差フォーマットのいずれかが選択された色差成分を用い、この色差フォーマットの種類に応じて、上記色差成分のブロックの動き補償の計算方法が選択されることを特徴とする請求項11記載の動き補償方法。

【請求項13】 上記複数種類の色差フォーマットは、
4:2:0フォーマット、4:2:2フォーマット、
4:4:4フォーマットであることを特徴とする請求項12記載の動き補償方法。

【請求項14】 上記色差成分のブロックの動きベクトルは、画像の上記輝度成分及び色差成分と共に供給される画像の形状情報に応じて計算されたものであることを特徴とする請求項12記載の動き補償方法。

【請求項15】 上記輝度成分及び色差成分のブロックを含むマクロブロック内の各ブロックの動き補償の計算方法を選択するフラグを有し、

マクロブロック内の色差ブロックについての動き補償の計算方法を、上記色差フォーマットの種類と、上記フラグとから選択することを特徴とする請求項12記載の動き補償方法。

【請求項16】 上記マクロブロックについて複数の予測モードが選択可能であり、上記動き補償の計算方法を、上記色差フォーマットの種類と、上記マクロブロックの予測モードとに応じて切換選択することを特徴とする請求項15記載の動き補償方法。

【請求項17】 上記色差フォーマットが4:2:2のとき、色差成分のブロックに対して、上下の隣接ブロックの動きベクトルを用いたオーバーラップ動き補償を施すことを特徴とする請求項12記載の動き補償方法。

【請求項18】 画像の輝度成分と色差成分とをそれぞれブロック化して各ブロック毎に求められた動きベクトルに基づいて動き補償を行う動き補償装置において、上記色差成分として、複数種類の色差フォーマットのいずれかが選択された色差成分を用い、

上記色差成分のブロックの動きベクトルは、上記輝度成分のブロックに対して検出された動きベクトルに基づいて生成されたものであり、

上記色差成分のブロックの動きベクトルに応じて当該色差成分のブロックの動き補償を行う際に、上記複数種類の色差フォーマットに応じた動き補償の計算方法を選択することを特徴とする動き補償装置。

【請求項19】 上記複数種類の色差フォーマットは、
4:2:0フォーマット、4:2:2フォーマット、
4:4:4フォーマットであることを特徴とする請求項10
18記載の動き補償装置。

【請求項20】 上記色差成分のブロックの動きベクトルは、画像の上記輝度成分及び色差成分と共に供給される画像の形状情報に応じて計算されたものであることを特徴とする請求項18記載の動き補償装置。

【請求項21】 上記輝度成分及び色差成分のブロックを含むマクロブロック内の各ブロックの動き補償の計算方法を選択するフラグを有し、

マクロブロック内の色差ブロックについての動き補償の計算方法を、上記色差フォーマットの種類と、上記フラグどから選択することを特徴とする請求項18記載の動き補償方法。20

【請求項22】 上記マクロブロックについて複数の予測モードが選択可能であり、上記動き補償の計算方法を、上記色差フォーマットの種類と、上記マクロブロックの予測モードとに応じて切換選択することを特徴とする請求項21記載の動き補償方法。

【請求項23】 上記色差フォーマットが4:2:2のとき、色差成分のブロックに対して、上下の隣接ブロックの動きベクトルを用いたオーバーラップ動き補償を施すことを特徴とする請求項21記載の動き補償装置。

【請求項24】 画像の輝度成分と色差成分とをそれぞれブロック化して各ブロック毎の動きベクトルを求め、動き補償を伴う予測符号化を行うことにより得られた符号化データを提供する提供媒体において、

上記符号化データは、

上記色差成分として、複数種類の色差フォーマットのいずれかが選択された色差成分を用い、

上記輝度成分のブロックの動きベクトルを検出すると共に、得られた輝度成分の動きベクトルに基づき、上記色差成分の色差フォーマットの種類に応じて選択された計算により、対応する色差成分のブロックの動きベクトルを生成し、

この動きベクトル検出手段により得られた動きベクトルを用いて動き補償を行い、

この動き補償手段からの出力と入力画像とに基づいて予測符号化を行うことにより得られたものであることを特徴とする符号化データの提供媒体。

【請求項25】 上記複数種類の色差フォーマットは、
4:2:0フォーマット、4:2:2フォーマット、

4:4:4フォーマットであることを特徴とする請求項24記載の提供媒体。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】 本発明は、動きベクトル生成方法、画像符号化装置、動き補償方法、動き補償装置、及び提供媒体に関し、特に、例えば動画像データを光磁気ディスクや磁気テープなどの記録媒体に記録し、これを再生してディスプレイなどに表示したり、テレビ会議システム、テレビ電話システム、放送用機器、マルチメディアデータベース検索システムなどのように、動画像データを伝送路を介して送信側から受信側に伝送し、受信側において、受信された動画像データを表示したり、編集して記録するための画像符号化等に適用可能な動きベクトル生成方法、画像符号化装置、動き補償方法、動き補償装置、及び提供媒体に関する。

【0002】

【従来の技術】 例えば、テレビ会議システム、テレビ電話システムなどのように、動画像データを遠隔地に伝送するシステムにおいては、伝送路を効率良く利用するために、画像のライン相関やフレーム間相関を利用して画像データを圧縮符号化するようにしている。

【0003】 動画像の高能率符号化方式には、代表的なものとしてMPEG (Moving Picture Experts Group) (蓄積用動画像符号化) 方式がある。これはISO-IEC/JTC1/SC2/WG11において議論され、標準案として提案されたものであり、動き補償予測符号化とDCT (Discrete Cosine Transform) 符号化を組み合わせたハイブリッド方式が採用されている。

【0004】 MPEGでは、様々なアプリケーションや機能に対応するために、幾つかのプロファイルおよびレベルが定義されている。最も基本となるのが、メインプロファイルメインレベル (MP@ML : Main Profile at Main Level) である。

【0005】 図42は、MPEG方式におけるMP@MLのエンコーダーの一般的な構成例を示している。

【0006】 符号化すべき入力画像データは、フレームメモリ31に入力され、一時記憶される。

【0007】 動きベクトル検出器32は、フレームメモリ31に記憶された画像データを、例えば16画素×16画素などで構成されるマクロブロック単位で読み出し、その動きベクトルを検出する。

【0008】 ここで、動きベクトル検出器32においては、各フレームの画像データを、Iピクチャ (フレーム内符号化)、Pピクチャ (前方予測符号化)、またはBピクチャ (両方向予測符号化) のうちの何れかとして処理する。なお、シーケンシャルに入力される各フレームの画像を、I, P, Bピクチャのいずれのピクチャとして処理するかは、例えば予め定められている (例えば、I, B, P, B, P, ..., B, P) として処理され

る)。

【0009】すなわち、動きベクトル検出器32は、フレームメモリ31に記憶された画像データの中の、予め定められた所定の参照フレームを参照し、その参照フレームと、現在符号化の対象となっているフレームの16画素×16ラインの小ブロック（マクロブロック）とをパターンマッチング（ブロックマッチング）することにより、そのマクロブロックの動きベクトルを検出する。

【0010】ここで、MPEGにおいて、画像の予測モードには、イントラ符号化（フレーム内符号化）、前方予測符号化、後方予測符号化、両方向予測符号化の4種類があり、Iピクチャはイントラ符号化され、Pピクチャはイントラ符号化または前方予測符号化の何れかで符号化され、Bピクチャはイントラ符号化、前方予測符号化、後方予測符号化、または両方法予測符号化の何れかで符号化される。

【0011】このため、動きベクトル検出器32は、Iピクチャについては予測モードとしてイントラ符号化モードを設定する。この場合、動きベクトル検出器32は、動きベクトルの検出は行わず、その予測モード（イントラ予測モード）の情報をVLC（可変長符号化）器36および動き補償器42に出力する。

【0012】また、動きベクトル検出器32は、Pピクチャについては前方予測を行い、その動きベクトルを検出する。さらに、動きベクトル検出器32は、前方予測を行うことにより生じる予測誤差と、符号化対象のマクロブロック（Pピクチャのマクロブロック）の、例えば分散とを比較する。その比較の結果、マクロブロックの分散の方が予測誤差より小さい場合、動きベクトル検出器32は、予測モードとしてイントラ符号化モードを設定し、検出した動きベクトルとともにそのモードの情報をVLC器36および動き補償器42に出力する。また、動きベクトル検出器32は、前方予測を行うことにより生じる予測誤差の方が小さければ、予測モードとして前方予測符号化モードを設定し、検出した動きベクトルとともにそのモードの情報を、VLC器36および動き補償器42に出力する。

【0013】さらに、動きベクトル検出器32は、Bピクチャについては、前方予測、後方予測、および両方向予測を行い、それぞれの動きベクトルを検出する。そして、動きベクトル検出器32は、前方予測、後方予測、および両方向予測についての予測誤差の中の最小のもの（以下、適宜、最小予測誤差という）を検出し、その最小予測誤差と、符号化対象のマクロブロック（Bピクチャのマクロブロック）の、例えば分散とを比較する。その比較の結果、マクロブロックの分散の方が最小予測誤差より小さい場合、動きベクトル検出器32は、予測モードとしてイントラ符号化モードを設定し、検出した動きベクトルとともにそのモードの情報を、VLC器36および動き補償器42に出力する。また、動きベクトル検

出器32は、最小予測誤差の方が小さければ、予測モードとして、その最小予測誤差が得られた予測モードを設定し、検出した動きベクトルとともにそのモードの情報を、VLC器36および動き補償器42に出力する。

【0014】動き補償器42は、動きベクトル検出器32から予測モードと動きベクトルの両方を受信すると、その予測モードおよび動きベクトルにしたがって、フレームメモリ41に記憶されている、符号化され且つ既に局所復号された画像データを読み出し、この読み出された画像データを予測画像データとして演算器33および40に供給する。

【0015】演算器33は、動きベクトル検出器32がフレームメモリ31から読み出した画像データと同一のマクロブロックをフレームメモリ31から読み出し、そのマクロブロックと、動き補償器42からの予測画像との差分を演算する。この差分値はDCT器34に供給される。

【0016】一方、動き補償器42は、動きベクトル検出器32から予測モードのみを受信した場合、すなわち予測モードがイントラ符号化モードである場合には、予測画像を出力しない。この場合、演算器33（演算器40も同様）は、特に処理を行わず、フレームメモリ31から読み出したマクロブロックをそのままDCT器34に出力する。

【0017】DCT器34では、演算器33の出力データに対してDCT処理を施し、その結果得られるDCT係数を、量子化器35に供給する。量子化器35では、バッファ37のデータ蓄積量（バッファ37に記憶されているデータの量）（バッファファイードバック）に対応して量子化ステップ（量子化スケール）が設定され、その量子化ステップでDCT器34からのDCT係数を量子化する。この量子化されたDCT係数（以下、適宜、量子化係数という）は、設定された量子化ステップとともにVLC器36に供給される。

【0018】VLC器36では、量子化器35より供給される量子化係数を、例えばハフマン符号などの可変長符号に変換し、バッファ37に出力する。さらに、VLC器36は、量子化器35からの量子化ステップ、動きベクトル検出器32からの予測モード（イントラ符号化（画像内予測符号化）、前方予測符号化、後方予測符号化、または両方向予測符号化のうちの何れが設定されたかを示すモード）および動きベクトルも可変長符号化し、その結果得られる符号化データをバッファ37に出力する。

【0019】バッファ37は、VLC器36からの符号化データを一時蓄積することにより、そのデータ量を平滑化し、符号化ビットストリームとして例えば伝送路に出力、または記録媒体に記録する。

【0020】また、バッファ37は、そのデータ蓄積量を量子化器35に出力しており、量子化器35は、この

バッファ37からのデータ蓄積量にしたがって量子化ステップを設定する。すなわち、量子化器35は、バッファ37がオーバーフローしそうなとき、量子化ステップを大きくし、これにより量子化係数のデータ量を低下させる。また、量子化器35は、バッファ37がアンダーフローしそうなとき、量子化ステップを小さくし、これにより量子化係数のデータ量を増大させる。このようにして、バッファ37のオーバーフローとアンダーフローを防止するようになっている。

【0021】量子化器35が outputする量子化係数と量子化ステップは、VLC器36だけでなく、逆量子化器38にも供給されるようになされている。逆量子化器38では、量子化器35からの量子化係数を、同じく量子化器35からの量子化ステップにしたがって逆量子化する。これにより、量子化係数はDCT係数に変換される。このDCT係数は、IDCT器(逆DCT器)39に供給される。IDCT器39では、DCT係数を逆DCT処理し、その処理の結果得られるデータを演算器40に供給する。

【0022】演算器40には、IDCT器39の出力データの他、上述したように動き補償器42から演算器33に供給されている予測画像と同一のデータが供給されている。演算器40は、IDCT器39の出力データ(予測残差(差分データ))と、動き補償器42からの予測画像データとを加算することで、元の画像データを局所復号し、この局所復号された画像データ(局所復号画像データ)を出力する。但し、予測モードがINTL(符号化)である場合、IDCT器39の出力データは演算器40をスルーして、そのまま局所復号画像データとしてフレームメモリ41に供給される。なお、この復号画像データは、受信側において得られる復号画像データと同一のものである。

【0023】演算器40において得られた復号画像データ(局所復号画像データ)は、フレームメモリ41に供給されて記憶され、その後、インター符号化(前方予測符号化、後方予測符号化、量方向予測符号化)される画像に対する参照画像データ(参照フレーム)として用いられる。

【0024】次に、図43は、図42のエンコーダから出力される符号化データを復号する、MPEGにおけるMP@MLのデコーダの一例の構成を示している。

【0025】このデコーダにおいて、バッファ101には、伝送路を介して伝送されてきた符号化ビットストリーム(符号化データ)を図示せぬ受信装置で受信し、または記録媒体に記録された符号化ビットストリーム(符号化データ)を図示せぬ再生装置で再生した、符号化ビットストリーム(符号化データ)が供給される。当該バッファ101は、この符号化ビットストリームを一時記憶する。

【0026】IVLC器(逆VLC器(可変長復号

器))102は、バッファ101に記憶された符号化データを読み出し、可変長復号することにより、その符号化データをマクロブロック単位で、動きベクトル、予測モード、量子化ステップ、および量子化係数に分離する。これらのデータのうち、動きベクトルおよび予測モードは動き補償器107に供給され、量子化ステップおよびマクロブロックの量子化係数は逆量子化器103に供給される。

【0027】逆量子化器103は、IVLC器102より供給されたマクロブロックの量子化係数を、同じくIVLC器102より供給された量子化ステップにしたがって逆量子化し、その結果得られるDCT係数をIDCT器104に出力する。IDCT器104は、逆量子化器103からのマクロブロックのDCT係数を逆DCTし、演算器105に供給する。

【0028】演算器105には、IDCT器104の出力データの他、動き補償器107の出力データも供給されている。すなわち、動き補償器107は、フレームメモリ106に記憶されている、既に復号された画像データを、図42の動き補償器42における場合と同様に、IVLC器102からの動きベクトルおよび予測モードにしたがって読み出し、予測画像データとして演算器105に供給する。演算器105は、IDCT器104の出力データ(予測残差(差分値))と、動き補償器107からの予測画像データとを加算することで、元の画像データを復号する。この復号画像データは、再生画像データとして出力されると共に、フレームメモリ106に供給されて記憶される。なお、IDCT器104の出力データがINTL(符号化)されたものである場合、その出力データは、演算器105をスルーして、復号画像データとして、そのままフレームメモリ106に供給されて記憶される。

【0029】フレームメモリ106に記憶された復号画像データは、その後に復号される画像データの参照画像データとして用いられる。さらに、復号画像データは、出力再生画像として例えば図示せぬディスプレイなどに供給されて表示される。

【0030】なお、MPEG1およびMPEG2では、Bピクチャは参照画像データとして用いられないため、エンコーダまたはデコーダのそれぞれにおいて、フレームメモリ41(図42)または106(図43)には記憶されない。

【0031】

【発明が解決しようとする課題】以上の図42、図43に示したエンコーダ、デコーダはMPEG1及びMPEG2の規格に準拠したものであるが、現在は、ISO-IEC/JTC1/SC29/WG11において、画像を構成する物体などのオブジェクトのシーケンスであるビデオオブジェクト(VO: Video Object)単位で符号化を行う方式についてMPEG(Moving Picture Exper

ts Group) 4としての標準化作業が進められている。

【0032】ところで、MPEG4では、現在4:2:0フォーマットと呼ばれる、図44に規定されている画像フォーマットのみが符号化/復号化できる画像として規定されている。

【0033】当該4:2:0フォーマットは、図44に示すように、輝度Y及び2つの色差Cr, Cbの信号で構成される。

【0034】4:2:0フォーマットは、輝度Yの走査線2本、および水平方向の2画素に対して、色差Cr, Cbがそれぞれ1画素づつ割り当てられる画像フォーマット（すなわち、輝度Yの4画素に対し色差Cr, Cbがそれぞれ1画素づつ割り当てられる）であり、その色差Cr, Cbの位置は輝度Yに対し同位置に存在する。

【0035】なお、輝度Yに対しての色差Cr, Cbの位置は、図44の位置だけではなく、使用される装置等によって異なる。

【0036】4:2:0フォーマットは、前述のとおり、輝度Yの4画素に対し色差Cr, Cbがそれぞれ1画素割り当てられているため、輝度Yに対し、色差Cr, Cbの解像度が低くなる。

【0037】そのため、放送局等で求められる画質の高い画像に対しては、その使用用途に応じて、4:2:0フォーマットではなく、4:2:2フォーマットや4:4:4フォーマットが用いられる。

【0038】上記4:2:2フォーマットは、図45に示す画素配置のように、輝度Yの走査線1本及び水平方向の2画素に対し、色差Cr, Cbがそれぞれ1画素ずつ（すなわち輝度Yの2画素に対し色差Cr, Cbが1画素）割り当てられる。

【0039】上記4:4:4フォーマットは、図46に示す画素配置のように、輝度Yの走査線1本及び水平方向の1画素に対し、色差Cr, Cbがそれぞれ1画素ずつ（すなわち、輝度Y、色差Cr, Cbの画素数、位置は等しい）が割り当てられる。

【0040】このように4:2:2フォーマットや4:4:4フォーマットは、4:2:0フォーマットに対し、色差信号の画素数が多く、画質を求める画像に対して使用することはその画質を向上させることに対し大変有効である。

【0041】しかし、MPEG4においては、前述したとおり4:2:0フォーマットの画像のみが規定されており、4:2:2フォーマットや4:4:4フォーマットをその符号化・復号画像として使用することは不可能である。

【0042】また、MPEG4では、画像の符号化だけではなく、形状情報も符号化できるように規格化されている。さらに、形状情報を持った画像を符号化する方法は、4:2:0フォーマットのみに対応しており、4:2:2フォーマット、4:4:4フォーマットには対応

していない。

【0043】そこで、本発明はこのような状況に鑑みてなされたものであり、形状情報を所有している画像においての4:2:2フォーマット、4:4:4フォーマットの色差成分の符号化、及び、MPEG4において4:2:2フォーマット、4:4:4フォーマットが使用できるように拡張された画像符号化等に適用可能であり、特に、色差ブロックの動きベクトル検出や動き補償に用いて好適な動きベクトル生成方法、画像符号化装置、動き補償方法、動き補償装置、及び提供媒体を提供することを目的とする。

【0044】

【課題を解決するための手段】本発明に係る動きベクトル生成方法は、ことにより、上述した課題を解決する。

【0045】VOLレイヤもしくはさらに上位のレイヤにおいて、色差フォーマットを示すフラグ(croma_type)を採用する。このフラグ(croma_type)は、2ビットのフラグであり、例えばその値を表1と定義することによって符号化/復号化を行う画像の色差フォーマットを特定が可能となる。

【0046】また、マクロブロックレイヤにおいても、従来、MPEG4では色差のブロックの符号化状態を表すフラグ(cbpc)を用いてその色差ブロックにおけるDCT係数の存在を判断している。MPEG4では、このフラグ(cbpc)をI-VOP、P-VOPのマクロブロックの場合には、VLCである(mcbpc)より得て、B-VOPの場合には、CBPBの下位2ビットから得ることができる。

【0047】本方式では、これを改良し、フラグ(croma_type)によりその読み込むビット量を変化させることが可能である。すなわちクロマフォーマット(croma_format)が4:2:0フォーマットを示す場合には従来通り2ビットのフラグ(cbpc)のみでそのブロックの存在を判断するが、さらにクロマフォーマット(croma_type)が4:2:0以外の色差フォーマットを示した場合、さらにその色差のタイプに応じて数ビット読み込みそのビットの状態に応じて色差ブロックのDCT係数の存在を判断することができる。

【0048】これらの手法により、MPEG4を4:2:0フォーマットだけではなく、4:2:2もしくは4:4:4フォーマットに拡張することが可能となる。

【0049】また、形状情報は輝度と同じ画素数を持つ、2値または多値の画像で構成され、色差の符号化/復号化時にこの形状情報を利用する際には、形状情報に特定の処理を施し、色差用の形状情報として利用している。そのため、本方式ではこの処理方法をクロマのフォーマットを示すフラグを用いて適用的に変化させる方法を提案し、それにより形状情報を有する4:2:2、4:4:4フォーマットの画像の符号化が可能になる。

【0050】

【発明の実施の形態】本発明の好ましい実施の形態について、図面を参照しながら説明する。

【0051】図1には本発明の画像符号化方法及び装置を適用したエンコーダの一実施の形態の構成例を示すが、この図1の構成について説明する前に、本発明を理解する上で必要な用語として、ビジュアルオブジェクト(Visual Object)、ビデオオブジェクト(Video Object)、以下適宜VOとする)、ビデオオブジェクトレイヤ(Video Object Layer、以下適宜VOLとする)、グループオブビデオオブジェクトプレーン(Group of Video Object Plane、以下適宜GOVとする)、ビデオオブジェクトプレーン(Video Object Plane、以下適宜VOPとする)、グループオブブロック(Group of block、以下適宜GOBとする)、マクロブロック(以下、適宜MBとする)、ブロック(block)について、図2及び図3を用いて簡単に説明する。

【0052】先ず、ブロックは、輝度または色差毎の隣り合った例えば8ライン×8画素から構成される。例えば、DCT(離散コサイン変換)はこの単位で実行される。なお、これ以降、各ブロックについては、便宜上図3の各ブロックに付加した番号により表すことにする。

【0053】4:2:0フォーマットにおけるマクロブロック内のブロックの配置を図3の(a)を用いて説明する。図3の(a)は、MPEG-4のビジュアルコミッティドラフト(Visual Committee Draft、以下、適宜、CDと記述する)における、マクロブロック内のブロックの分割法を示した図である。

【0054】マクロブロック(MB)は、例えば画像のフォーマットが、いわゆる4:2:0コンポーネントデジタル信号である場合、上下左右に隣り合った4つの輝度Yのブロックと、画像上では同じ位置にあたる色差Cb、Crそれぞれのブロックの全部で6つのブロックで構成される。すなわち、MPEG-4には、従来4:2:0フォーマットのマクロブロックをその符号化/復号画像として定義しており、図3の(a)のブロック0、1、2、3は輝度Yのブロックを示し、ブロック4は色差Cbのブロックを、ブロック5は色差Crのブロックを示す。各ブロックは8×8画素で構成されている。ここで、図3の(a)の輝度Yのブロックの配置は、16×16画素の輝度Yのブロックを4つの8×8のブロックに分割して考えることを示す。同様に色差Cb、Crのブロック4、5も8×8画素のブロックである。このとき4:2:0フォーマットでは、色差の大きさは一つのマクロブロックでそれぞれ8×8画素となされており、この色差のブロックは分割せずにそのまま符号化される。また、各マクロブロックを8×8画素のブロックに分割して考えるのは、MPEGにおいてDCTが8×8画素のブロックに対して行われるものであるからである。

【0055】ここで、本実施の形態では、色差フォーマ

ットが、4:2:0フォーマットだけでなく、4:2:2フォーマット、4:4:4フォーマットにも対応している。図3の(a)のブロックの配置では、4:2:2フォーマット、4:4:4フォーマットに対応できないため、4:2:2フォーマットに対応させたときのブロック配置を図3の(b)に、4:4:4フォーマットに対応させたときのブロック配置を図3の(c)に示す。なお、図3の(b)、図3の(c)の各ブロックは、図3の(a)と同様に、それぞれ8×8画素のブロックを示す。

【0056】図3の(b)は、4:2:2フォーマットに対応したマクロブロック内のブロックの構造を模式的に示している。図3の(b)のブロック配置では、輝度Yが4つのブロック0、1、2、3で構成され、色差Cb、Crのブロックは上下に位置するそれぞれ2つの8×8のブロック4と6、5と7で構成される。すなわち4:2:2フォーマットの輝度Yのブロックは、4:2:0フォーマットと同様16×16のブロックを上下左右に2等分づつした形になっているが、色差のブロックは8×16画素で構成されている。そのため、図3の(b)の様に、色差信号は上下2つに分割されている。

【0057】さらに、図3の(c)は、4:4:4フォーマットに対するマクロブロックの構造を模式的に示している。4:4:4フォーマットでは、輝度Y及び色差Cb、Crはそれぞれ16×16画素で構成される。そのため4:4:4フォーマットをブロックに分割した場合、Y、Cb、Crのそれぞれが図3の(c)に示すような4つの8×8ブロックで構成されることとなる。すなわち、輝度Yはブロック0、1、2、3で構成され、色差Cbのブロックは4、6、8、10で、色差Crのブロックは5、7、9、11で構成される。

【0058】次に、VOP(Video Object Plane)は、1つまたは複数のマクロブロックから構成される1枚の画像を表す。この画像は、符号化される方式にしたがって、Iピクチャ(画像内符号化画像)、Pピクチャ(前方予測符号化画像)、およびBピクチャ(両方向予測符号化画像)のうちのいずれかに分類される。

【0059】I-VOP(Iピクチャ)は、動き補償を行うことなく、画像そのものが符号化(インストラ符号化)されるものである。P-VOP(Pピクチャ)は、基本的には、自身より時間的に前に位置する画像(IまたはP-VOP)に基づいて、前方予測符号化される。B-VOP(Bピクチャ)は、基本的には、自身より時間的に前と後ろに位置する2つの画像(IまたはP-VOP)に基づいて両方向予測符号化されるものである。

【0060】GOV(Group of Video Object Plane)は、少なくとも1枚のI-VOPと、0または少なくとも1枚の非I-VOP(すなわちP-VOP、B-VOP)とから構成される。ただし、階層符号化を行った場合の上位レイヤに関してはこの限りではない。なお、G

VOの使用、不使用は符号化時に自由に決められる。

【0061】VOL (Video Object Layer) は、少なくとも1枚以上のVOPもしくはGOVから成り立つ。VOL内には同一の表示時間を持つVOPは含まれず、VOPをその表示時間順に表示することにより一連の画像が表示されることになる。

【0062】ビデオオブジェクト (video Object: VO) は、1つ以上のVOLから構成され、VOLの組合せにより同一オブジェクトを複数の解像度、フレームレート等で符号化、復号化することが可能である。これは図2に表す関係を有する。

【0063】ビジュアルオブジェクト (Visual Object) は、一つのビデオオブジェクト (video Object) または他の一つのオブジェクト (例えばフェイスオブジェクト (face object)、メッシュオブジェクト (mesh object)、スチルテクスチャオブジェクト (still texture object) 等) により構成される。

【0064】ビデオオブジェクトシーケンス (Visual Object Sequence) は、一つもしくは複数のビジュアルオブジェクト (Visual Object) より構成され、図2に表す関係を有する。

【0065】次に、図1を用いて、本発明を適用した実施の形態のエンコーダについて説明する。

【0066】入力画像信号すなわち符号化すべき画像 (動画像) データは、VO (Video Object) 構成部1に入力される。VO構成部1では、入力される画像を構成するオブジェクト毎に、そのシーケンスであるVO (Video Object) を構成し、VOP構成部21乃至2Nに出力する。すなわち、VO構成部1においてN個のVO#1乃至VO#Nが構成された場合、そのN個のVO#1乃至VO#Nは、それぞれ対応したN個のVOP構成部21乃至2Nにそれぞれ出力される。

【0067】図4を用いて具体的に説明すると、符号化すべき画像データが、例えば独立した背景F1のシーケンスと前景F2のシーケンスとから構成される場合、VO構成部1は、例えば背景F1のシーケンスをVO#1としてVOP構成部21に出力するとともに、前景F2のシーケンスをVO#2としてVOP構成部22に出力する。なお、図4の例は、背景である画像F1と前景である画像F2どちらなる画像を示しており、画像F1は、例えばある自然の風景を撮影したものであり、その画像全体のシーケンスが1つのVO (VO#1) とされている。また、画像F2は、例えば人が歩いている様子を撮影したものであり、その人を囲む最小の長方形のシーケンスが1つのVO (VO#2) とされている。

【0068】VO構成部1は、符号化すべき画像データが、例えば背景F1と前景F2とが既に合成されたものである場合、所定のアルゴリズムにしたがって当該画像を領域分割することにより、背景F1と前景F2とを取り出し、それぞれのシーケンスとしてのVOを、対応す

るVOP構成部2n (但し、n=1, 2, ..., N) に出力する。

【0069】VOP構成部2nは、VO構成部1の出力からVOPを構成する。すなわち例えば、各フレームから物体 (オブジェクト) を抽出し、その物体を囲む、例えば最小の長方形 (以下、適宜、最小長方形という) をVOPとする。なおこのとき、VOP構成部2nは、その横および縦の画素数が、例えば16の倍数となるようにVOPを構成する。VO構成部2nは、VOPを構成すると、そのVOPをVOP符号化部3nに出力する。

【0070】さらに、VOP構成部2nは、VOPの大きさ (例えば横および縦の長さ) を表すサイズデータ (VOP_size) と、フレームにおけるそのVOPの位置 (例えばフレームの最も左上を原点とするときの座標) を表すオフセットデータ (VOP_offset) とを検出し、これらのデータもVOP符号化部3nに供給する。

【0071】VOP符号化部3nは、VOP構成部2nの出力を、例えばMPEGやH.263などの規格に準拠した方式で符号化し、その結果得られるビットストリームを多重化部4に出力する。多重化部4は、VOP符号化部31乃至3Nからのビットストリームを多重化し、その結果得られる多重化データを、例えば地上波や衛星回線、CATV網その他の伝送路5を介して伝送し、または、例えば磁気ディスク、光磁気ディスク、光ディスク、磁気テープその他の記録媒体6に記録する。

【0072】ここで、VOおよびVOPについて説明する。

【0073】VOは、ある合成画像のシーケンスが存在する場合の、その合成画像を構成する各オブジェクト (物体) のシーケンスであり、VOPは、ある時刻におけるVOを意味する。すなわち、例えば画像F1およびF2を合成して構成された合成画像F3がある場合、画像F1またはF2が時系列に並んだものがそれぞれVOであり、ある時刻における画像F1またはF2がそれぞれVOPである。従って、VOは、異なる時刻の同一オブジェクトのVOPの集合であると言うことができる。

【0074】なお、図4の例のように、画像F1を背景とするとともに画像F2を前景とすると、それらの合成画像F3は、画像F2を切り抜くためのキー信号を用いて、画像F1およびF2を合成することによって得られるが、この場合における画像F2のVOPには、その画像F2を構成する画像データ (輝度信号および色差信号) の他、適宜、そのキー信号も含まれるものとする。

【0075】画像フレーム (画枠) のシーケンスは、その大きさおよび位置のいずれも変化しないが、VOは、大きさや位置が変化する場合がある。すなわち、同一のVOを構成するVOPであっても、時刻によってその大きさや位置が異なる場合がある。

【0076】具体的にいって、図4の例のように、画像

F 1 が例えはある自然の風景を撮影したものであり、画像 F 2 が例えは人が歩いている様子を撮影したものである場合、画像 F 1 の VOP # 1 は風景の画像であるから、基本的に通常の画像のフレームと同様にその位置および大きさの両方とも変化しないのに対して、画像 F 2 の VOP # 2 は人の画像であるから、人物が左右に移動したり、また図面において手前側または奥側に移動することにより、その大きさや位置が変化する。従って、図 4 は、同一時刻における VOP # 1 および VOP # 2 を表しているが、VOP の位置や大きさは時間の経過に伴って変化することがある。

【0077】そこで、図 1 の VOP 符号化部 3n は、その出力するビットストリームに、VOP を符号化したデータの他、所定の絶対座標系における VOP の位置（座標）および大きさに関する情報も含めるようになされている。なお、図 4 においては、VOP # 1 を構成するある時刻の VOP（画像 F 1）の位置を示すベクトルを OST 1 とし、その時刻と同一時刻における VOP # 2 の VOP（画像 F 2）の位置を示すベクトルを OST 2 として、それぞれ表してある。

【0078】次に、図 5 には、図 1 の VOP 符号化部 3n の詳細な構成例を示す。なお、この図 5 において、図 4 2 における場合と対応する各構成要素については、同一の指示符号を付してある。すなわち、VOP 符号化部 3n は、基本的には図 4 2 のエンコーダと同様に構成されているものである。

【0079】この図 5 において、図 4 2 における場合と同様に、入力画像の画像データはフレームメモリ 31 に供給されて VOP として記憶される。

【0080】動きベクトル検出器 32 では、フレームメモリ 31 に記憶された VOP に対してマクロブロック単位で動きベクトルの検出を行う。すなわち、上述したように、VOP は時刻（フレーム）によって大きさや位置が変化するため、その動きベクトルの検出にあたっては、その検出のための基準となる座標系を設定し、その座標系における動きを検出する必要がある。そこで、動きベクトル検出器 32 では、上述の絶対座標系を基準となる座標系とし、サイズデータ FSZ_B およびオフセットデータ FPOS_B にしたがってその絶対座標系に符号化対象の VOP 及び参照画像とする VOP を配置して、動きベクトルを検出する。なお、検出された動きベクトル（MV）は、予測モードとともに形状情報符号化器 50、VLC 器 36 および動き補償器 42、量子化器 35、逆量子化器 38、DCT 係数差分化器 44 に供給される。

【0081】また、動き補償を行う場合においても、やはり上述したように基準となる座標系における動きを検出する必要があるため、動き補償器 42 には、サイズデータ FSZ_B およびオフセットデータ FPOS_B が供給されるようになされている。

【0082】演算器 33 には、図 4 2 の場合と同様に、動きベクトル検出器 32 がフレームメモリ 31 から読み出した画像データにおけるものと同一のマクロブロックのデータが供給される。この演算器 33 では、そのマクロブロックと動き補償器 42 からの予測画像との差分を演算する。この差分値は、DCT 器 34 に送られる。動き補償器 42 は、図 4 2 の場合と同様に、予測モードがイントラ符号化モードである場合には、予測画像を出力しない。この場合、演算器 33（演算器 40 も同様）は、特に処理を行わず、フレームメモリ 31 から読み出したマクロブロックのデータをそのまま DCT 器 34 に出力する。

【0083】DCT 器 34 では、演算器 33 の出力データに対して、8 ライン × 8 画素からなるブロック単位で DCT 处理を施し、その DCT 处理の結果得られる DCT 係数を量子化器 35 に供給する。

【0084】量子化器 35 では、図 4 2 の量子化器 35 と同様に、入力された DCT 係数を量子化し、その量子化データを DCT 係数差分化器 44 及び逆量子化器 38 に送る。

【0085】逆量子化器 38 では、図 4 2 に示す逆量子化器 38 と同様の動作を行う。すなわち、量子化器 35 から入力された 8 × 8 の量子化された DCT 係数を逆量子化し、IDCT 器 39 に送る。

【0086】IDCT 器 39 は、図 4 2 の IDCT 器 39 と同様に動作し、逆量子化器 38 より逆量子化された DCT 係数を、IDCT 处理して演算器 40 に出力する。

【0087】演算器 40 には、IDCT 器 39 の出力データの他、動き補償器 42 から、演算器 33 に供給されている予測画像と同一のデータが供給されている。演算器 40 は、IDCT 器 39 の出力データ（予測残差（差分データ））と、動き補償器 42 からの予測画像データとを加算することで、元の画像データを局所復号し、この局所復号した画像データ（局所復号画像データ）を出力する。但し、予測モードがイントラ符号化である場合には、IDCT 器 39 の出力データは演算器 40 をスルーして、そのまま局所復号画像データとしてフレームメモリ 41 に供給される。なお、この復号画像データは、受信側において得られる復号画像データと同一のものである。

【0088】演算器 40 において得られた復号画像データ（局所復号画像データ）は、テクスチャ（texture）情報として後述するパディング（Padding）処理器 51 に供給される。

【0089】一方、形状情報（キー信号）、入力されたサイズデータ FSZ_B、オフセットデータ FPOS_B、VOP のサイズデータ VOP_size、VOP のオフセットデータ VOP_offset 及び、動きベクトル検出器 32 より出力された動きベクトルと予測モードは、形状

情報符号化器50に供給される。形状情報符号化器50では、MPEG4のビジュアルコミッティドラフト(Visual Committee draft)にある記述に従ってその形状情報の符号化を行う。符号化された形状情報は、形状情報復号器52及びVLC器36へ送られる。

【0090】形状情報復号器52は、形状情報符号化器50から供給された、符号化されている形状情報の局所復号化を行い、そのデータを色差情報用作成器53及びパディング(Padding)処理器51、DCT係数差分化器44及び、VLC器36へ送る。

【0091】色差用形状情報作成器53では、色差フォーマットの形式により形状情報へ処理を加える。当該色差用形状情報作成器53の動作について以下に説明する。図6には、4:2:0フォーマット(図6の(a))、4:2:2フォーマット(図6の(b))、4:4:4フォーマット(図6の(c))の場合の色差用形状情報の作成方法を説明するための図を示す。

【0092】4:2:0フォーマットの場合を示す図6の(a)において、輝度Yの画素1a, 1b, 1c, 1dのと色差Cb, Crの画素ca(図6の(a)では同じ位置に色差Cb及びCrが存在する)が対応した位置に存在する画素となり、同様に輝度の画素1e, 1f, 1g, 1hと色差の画素cbが対応した位置に存在する画素となる。すなわちここで対応とは1aの位置の画素を実際に表示しようとした場合、当該1aの位置にある輝度Yの画素値、及びcaの位置にある色差Cb, Crの画素値を用いてその表示する画素の情報として用いることである。

【0093】ここで、4:2:0フォーマットの場合、次段に送られることになる形状情報は、輝度の画素と一対一に対応するように同数で同位置の情報のみとなれる。すなわち、図6の(a)の場合は、輝度の画素1a, 1b, 1c, 1d, 1e, 1f, 1g, 1hの位置に対応する形状情報のみであり、色差の位置に対応した形状情報は送られない。そのため、色差信号の符号化をする際には、色差の画素と一対一対応した同位置で同数の色差用の形状情報を、輝度用の形状情報より作成する必要がある。具体的にいうと、色差の画素caの位置(この位置にはCb, Crの色差情報が存在する)に対応する輝度の画素は1a, 1b, 1c, 1dであり、したがって色差の画素caの位置に対応する色差用の形状情報は、これら1a, 1b, 1c, 1dの位置に存在する形状情報より作成することになる。

【0094】形状情報は、通常2値の画像で伝送され、例えば値が0のとき当該画素はオブジェクトの外側であることを、値が1のとき当該画素がオブジェクトの内側であることを示している。そのため、4:2:0フォーマットにおいて、対応する輝度の画素の形状情報の中に一つでも形状情報が1となっている画素、すなわちオブジェクトの内側であると判断された画素の場合には、対

応する色差の形状情報も1(オブジェクトの内側)であるとされる。また、対応する輝度の画素が全て0(オブジェクトの外側)である場合、色差の画素も0(オブジェクトの外側)となる。例えば、輝度の画素1a, 1b, 1c, 1dの形状情報が1、すなわち当該画素がオブジェクトの内側である場合には、caに位置する色差用の形状情報も1、すなわちオブジェクトの内側であるとして判断され、逆に、輝度の画素1a, 1b, 1c, 1dの形状情報が0、すなわち当該画素がオブジェクトの外側である場合には、caに位置する色差用の形状情報も0、すなわちオブジェクトの外側であるとして判断される。なお、グレイスケールシェイプ(gray-scale shape)の場合の形状情報としては、上記2値情報に加え、形状情報の階調を表す情報が別途送られる。

【0095】次に、4:2:2フォーマットの場合は、図6の(b)に示すように輝度の画素1a, 1bと色差の画素ca(この位置には色差情報Cb, Crが存在する)が対応し、同様に輝度の画素1c, 1dと色差の画素cbが対応し、輝度の画素1e, 1fと色差の画素ccが、輝度の画素1g, 1hと色差の画素cdがそれぞれ対応する。4:2:2フォーマットの場合は、対応する2つの輝度の画素の両方または何れか一方の形状情報が1、すなわちオブジェクトの内側であることを示している場合、対応する色差の形状情報も1(オブジェクトの内側)となる。また、対応する2つの輝度の画素の何れも形状情報が0(オブジェクトの外側)である場合、同位置の色差の形状情報も0(オブジェクトの外側)となる。例えば、図6の(b)の場合には、輝度の画素1a, 1bの位置に属する形状情報のどちらかまたは両方が1の場合、色差の画素caの位置に属する形状情報も1となる。また輝度の画素1a, 1bの位置に属する形状情報がどちらも0の場合、色差の画素caの位置に属する形状情報も0となる。

【0096】次に、4:4:4フォーマットの場合は、図6の(c)に示すように、各輝度の画素と色差の画素が同位置に配置される。すなわち、輝度の画素1aと色差の画素ca(この位置には色差情報Cb, Crが存在する)が対応し、以下同様に、輝度の画素1bと色差の画素cbが、輝度の画素1cと色差の画素ccが、輝度の画素1dと色差の画素cdが、輝度の画素1eと色差の画素ceが、輝度の画素1fと色差の画素cfが、輝度の画素1gと色差の画素cgが、輝度の画素1hと色差の画素chがそれぞれ対応する。したがって、この4:4:4フォーマットの場合は、輝度の画素1aの位置に属する形状情報はそのまま色差の画素caの位置にある色差の画素(Cb, Cr)の形状情報として活用することが可能である。他の色差の画素cb, cc, cd, ce, cf, cg, chについても同様に、それぞれ対応する輝度の画素1b, 1c, 1d, 1e, 1f, 1g, 1hの位置に属する形状情報を活用する。

【0097】このようにして 色差用形状情報作成器53では、輝度の画素と同位置にある形状情報より、その符号化を行っている色差フォーマットに合わせて色差用の形状情報を作成する。当該色差用形状情報作成器53にて作成された色差用形状情報は、パディング(Padding)処理器51、DCT係数差分化器44、VLC器36に供給される。

【0098】ここで、図7及び図8を用いて、パディング処理器51の動作を説明する。図7は、パディング処理器51の具体的な構成例を示している。

【0099】図7において、図5の演算器40からの局所復号画像データであるテクスチャ(texture)情報は、水平方向パディング処理器61に供給される。また、水平方向パディング処理器61には、図5の形状情報復号器52及び、色差用形状情報作成器53より、輝度のパディングに用いる形状情報、及び色差のパディングに用いる形状情報が供給される。

【0100】図8には、水平方向パディング処理器61にて行われる操作手順を示す。

【0101】パディング処理は、各マクロブロック毎に、また各マクロブロック内では輝度及び2つの色差情報毎に行われる。ここで、マクロブロックの輝度情報についてパディング処理を行う場合、テクスチャ(texture)情報の輝度の画素値(1.6画素×16画素)が図8に示す操作手順の中の変数d[y][x]として、また対応する位置の形状情報(16画素×16画素)が図8に示す操作手順中の変数s[y][x]として処理される。また、色差情報についてパディング処理を行う場合、それぞれテクスチャ情報が変数d[y][x]として、また色差用の形状情報が変数s[y][x]として処理される。ただし、色差情報の場合、その処理を行う画素数は4:2:0フォーマットの場合で8画素×8画素、4:2:2フォーマットの場合で8画素×16画素、4:4:4フォーマットの場合で16画素×16画素単位となる。

【0102】当該水平方向パディング処理器61でのパディング処理は、各ライン毎に行われる。すなわち、水平方向パディング処理器61では、マクロブロックの各ラインに対して順次パディング操作を行って、水平方向のパディングを行う。なお、水平方向のパディングを行うライン内にオブジェクト内の画素が一つも存在しない場合、そのラインについてはパディング処理が行われない。水平方向のラインに1つ以上のオブジェクトが存在している場合に、図8の処理が行われることとなる。

【0103】上記水平方向パディング処理器61にて水平方向にパディング処理が施されたテクスチャ(texture)情報と形状情報は、垂直方向パディング処理器62に供給される。この垂直方向パディング処理器62に供給されるテクスチャ情報は、図8に示す操作手順中の変数hor_pad[y][x]で表され、また同様に垂直方向パディング処理器62へ供給される形状情報は、図8に示す

操作手順の中の変数s'[y][x]で表される。また、図8に示す操作手順中の変数x'はパディング処理を行っている水平ラインにおいて、xの左側に位置するオブジェクトの境界の画素の位置を表し、変数x'は同様に右側のオブジェクトの境界の位置を示す。

【0104】図9には、垂直方向パディング処理器62にて行われる操作手順を示す。

【0105】この図9に示す操作手順中の変数hor_pad[y][x]は、上記水平方向にパディング処理を行ったテクスチャ情報であり、図9に示す操作手順中の変数s'[y][x]は水平方向のパディング時に変更された形状情報であり、どちらも水平方向パディング処理器61より供給されたものである。

【0106】この垂直方向パディング処理器62においても水平方向パディング処理器61と同様、色差情報の場合、その処理を行う画素数は、4:2:0フォーマットの場合で8画素×8画素、4:2:2フォーマットの場合で8画素×16画素、4:4:4フォーマットの場合で16画素×16画素単位となる。

【0107】また、この垂直方向パディング処理器62にて垂直方向にもパディング処理が行われたテクスチャ情報は、図9に示す操作手順中の変数hv_pad[y][x]で表される。また、図9に示す操作手順中の変数y'はパディング処理を行う垂直ラインにおいて、yの上側に位置するオブジェクトの境界の画素の位置を表し、図9に示す操作手順中の変数y''は同様に下側のオブジェクトの境界の位置を示す。

【0108】垂直方向パディング処理器62でのパディング処理は、各垂直ライン毎に行われる。すなわち、当該垂直方向パディング処理器62では、マクロブロックの各垂直ラインに対して順次パディング操作を行って、垂直方向のパディングを行う。なお、垂直方向のパディングを行うライン内にオブジェクト内の画素が一つも存在しない場合、そのラインについてはパディング処理が行われない。垂直方向のラインに1つ以上のオブジェクトが存在している場合に、図9で示す処理が行われる。図9に示す操作手順中の変数hv_pad[y][x]は、拡張パディング器63へ供給される。

【0109】拡張パディング器63では、オブジェクトの境界に位置するマクロブロックの上下左右に隣接するマクロブロックに対し、さらにパディング処理を行う。すなわち図10に示す例の場合、図中の黒色のマクロブロックがその対象となる。当該拡張パディングによるマクロブロック(previous block)は、例えば図11の(a)、(b)に示すように、隣接するオブジェクトの境界に位置するマクロブロック(current block)のその端の画素値を用いてパディングされる。また、図12に示すように、パディングされるマクロブロック(Exterior macroblock)が複数のオブジェクトの境界を持つマクロブロック(Boundary macroblock 0～Boundary ma

21

croblock 3) に囲まれている場合、この図 12 のマクロブロック (Boundary macroblock 0 ~ Boundary macroblock 3) に付けられている 0 ~ 3 の番号の順番に参照するマクロブロックを選択する。

【0110】拡張パディング器 63 の出力は、図 5 のフレームメモリ 41 に供給されて記憶され、その後、インター符号化 (前方予測符号化、後方予測符号化、量方向予測符号化) される画像に対する参照画像データ (参照フレーム) として用いられる。

【0111】そして、フレームメモリ 41 に記憶された画像データは、後方予測に用いる画像、または前方予測に用いる画像データとして、動き補償器 42 より出力されることになる。

【0112】一方、動き補償器 42 は、動き補償参照画像指示信号により指定される画像 (フレームメモリ 41 に記憶されている局所復号された画像) に対して、動きベクトル検出器 32 からの予測モードおよび動きベクトルをもとに動き補償を施し、予測画像を生成して演算器 33 および 40 に出力する。すなわち、動き補償器 42 は、前方/後方/両方向予測モードのときのみ、フレームメモリ 41 の読み出しアドレスを、演算器 33 に対して現在出力しているブロックの位置に対応する位置から動きベクトルに対応する分だけずらして、当該フレームメモリ 41 から前方予測または後方予測に用いる画像データを読み出し、予測画像データとして出力する。なお、両方向予測モードのときは、前方予測と後方予測に用いる画像データの両方が読み出され、例えばその平均値が予測画像データとして出力される。このようにして得られた予測画像データが、減算器としての前記演算器 33 に供給され、前述したように差分データが生成される。

【0113】ここで、図 13 を参照しながら、動き補償器 42 の画像の輝度成分と色差成分に対する動作について具体的に説明する。動き補償器 42 は各マクロブロック単位で動き補償を行う。図 13 は、マクロブロックに対する動き補償動作を示すフローチャートである。

【0114】本実施の形態の動き補償においては、動きベクトルを生成する際に、上記輝度成分のブロックに対して得られた動きベクトルに基づいて、対応する色差成分のブロックの動きベクトルを生成しており、このとき、上記色差成分の色差フォーマットが、4:2:0 フォーマット、4:2:2 フォーマット、4:4:4 フォーマットのいずれであるかに応じて、上記輝度成分のブロックに対して得られた動きベクトルに基づく色差成分のブロックの動きベクトルの計算方法を切換選択している。また、上記色差フォーマット、マクロブロックの動き予測モードのタイプ、及び形状情報に応じて、動き補償の計算方法を切換選択している。

【0115】先ず、図 13 の最初のステップ S201 では、動きベクトル予測モード、動きベクトル、形状情報

10

20

30

40

40

の読み込みを行う。次のステップ S202 で、予測されるブロックの決定を行い、次のステップ S203 で、全てのブロックが予測されているか否かを判別し、YES のときはステップ S201 に戻り、NO のときは次のステップ S204 に進んでいる。ステップ S204 では、現在のブロックが輝度 ("luminance") ブロックか否かを判別し、YES のときはステップ S211 に進み、NO のときは、ステップ S205 以降のクロマタイプ (chroma_type) に応じた処理に進んでいる。

【0116】ここで、上記ステップ S201 において読み込まれる動きベクトル検出器 32 からの予測モードとしては、後述する図 3.4 に示すように、マクロブロック毎に、イントラ (intra)、インター (inter)、インター+q (inter+q) 及びインター 4V (inter4v) の 4 種類のうちのいずれかの予測モードとなる。

【0117】マクロブロックのモードが、上記 4 種類の内のインター (inter) 及びインター+q (inter+q) モードの場合は、このマクロブロック MB に対してただ 1 つの動きベクトルが動きベクトル検出器 32 から得られ、このベクトルが図 3 における輝度ブロック 0 ~ 3 全てに対してそのまま動きベクトルとして用いられる。マクロブロックのモードが、インター 4V (inter4v) モードの場合は、このマクロブロック MB に対して、K ($1 < K \leq 4$) 個の動きベクトルが得られる。ここで K は、形状情報符号化器 50 からの形状情報から、マクロブロックの 4 個の輝度ブロックのうち、オブジェクトの内側にあると判断された輝度ブロックの個数である。これらのベクトルは、それぞれ、図 14 の (c)、図 15 の (c)、図 16 の (c) に示すように、オブジェクトの内側にある輝度ブロックの動きベクトルとしてそのまま用いられる。このように輝度成分に対しては、動きベクトル検出器 32 からの動きベクトルを変換せずに用いるが、画像の解像度とマクロブロック内のブロック数の異なる色差成分に対する動きベクトルは、輝度ブロックの動きベクトルを元に以下のように計算される。

【0118】マクロブロックが 4:2:0 フォーマットの場合には、図 13 のステップ S205 で YES (chroma_type = "01") と判別され、ステップ S206 で、4:2:0 色差動きベクトル計算が行われる。

【0119】この 4:2:0 フォーマットの場合において、色差ブロック 4, 5 の動きベクトルは、マクロブロックのモードがインター (inter) 又はインター+q (inter+q) モードの場合、図 14 の (a) に示すように、対応する輝度ブロックの動きベクトルの解像度を半分にしたベクトルとする。マクロブロックのモードがインター 4V (inter4v) の場合、図 14 の (b)、(c) に示すように、オブジェクトの内部にある K 個の輝度ブロックの動きベクトルを全て足し合わせ、その平均と解像度が半分であるということから、 $2 \times K$ で割り算することにより色差ブロックの動きベクトルが計算さ

50

れる。図14の(b)ではK=4、図14の(c)ではK=2の例が書かれている。

【0120】ここで、動きベクトル検出器32から得られる動きベクトルは、その解像度を2倍としたhalf sample resolutionを採用しており、色差ブロックの動きベクトル要素の解像度変換として、そのまま割り算を行うのではなく、図17に示すテーブルを用いる。例えば、K=4の場合は、動きベクトルMVを、sixteenth sample resolutionからhalf sample resolutionに変換するのに、図17の(a)を用いる。同様に、K=3、2、1の場合には、それぞれ、図17の(b)、(c)、(d)の変換を用いる。

【0121】このようにして計算された動きベクトルとフレームメモリーから得られる参照画像から、ステップS207での動き補償により、輝度ブロックと色差ブロックから構成される4:2:0予測マクロブロック画像が得られる。また、輝度ブロックの予測画像は、VOLシンタックス(図28)のフラグobmc_disable=0の場合、現在のブロックの1個の動きベクトルによる動き補償ではなく、オーバーラップ動き補償により得られる。

【0122】なお、輝度ブロックの動き補償については、図13のステップS211において、上記フラグobmc_disableが0か否かの判別がなされ、obmc_disable=0(YES)の場合にはオーバーラップ動き補償が行われ、NOの場合には、ステップS212に進んで通常の(オーバーラップ動き補償でない)動き補償が行われる。ここで、オーバーラップ動き補償について説明する。オーバーラップ動き補償は、obmc_disable=0の時のみ実行されるブロック単位の動き補償法であり、隣接ブロックと動きベクトルの相関が高い輝度ブロックに對しての適用される。オーバーラップ動き補償では、輝度ブロックの予測画像は、図18の(a)に示す、この予測ブロックの動きベクトルと、上下左右に隣接するブロックの動きベクトルMVを用いることにより得られる。

【0123】具体的には、予測ブロック内の各画素値は、現在の輝度ブロックの動きベクトルMVと2個の隣接ブロックの動きベクトルMVとを用いて得られた3個の予測画素値に、図19の(a)、(b)、(c)のmatrixに示す重みを掛けて足し合わせ、8で割ることにより得られる。例えば、予測輝度ブロックの上半分に存在する画素に対しては、このブロックの真上に隣接するブロックの動きベクトルMVが使われ、逆に下半分の画素に対しては、真下に隣接するブロックの動きベクトルMVが用いられる。同様に、このブロックの左半分の画素には、真左のブロックの動きベクトルMVが用いられ、右半分には、真右のブロックの動きベクトルMVが用いられる。この計算を各画素に対して一般化した式を、以下に示す。

【0124】すなわち、予測輝度ブロック内の各画素値をp(i,j)とすると、

$$p(i,j) = (q(i,j) \times I-I0(i,j) + r(i,j) \times I-I1(i,j) + s(i,j) \times I-I2(i,j) + 4) // 8$$

ここで、q(i,j)、r(i,j)、及びs(i,j)は、参照画像から得られる画素値であり、

$$q(i,j) = q(i+MV^i, j+MV^j),$$

$$r(i,j) = r(i+MV^i, j+MV^j),$$

$$s(i,j) = s(i+MV^i, j+MV^j)$$

である。また、ベクトル(MVⁱ, MV^j)は、現在のブロックの動きベクトルであり、ベクトル(MVⁱ, MV^j)は、上部、又は下部隣接ブロックの動きベクトル、ベクトル(MVⁱ, MV^j)は、左側、又は右側隣接ブロックの動きベクトルである。さらに、行列I-I0(i,j)、I-I1(i,j)、及びI-I2(i,j)は、図19の(a)、(b)、(c)で定義された行列である。

【0125】ただし、隣接ブロックが符号化されていない場合は、このブロックの動きベクトルは0とする。また、隣接ブロックに対応するマクロブロックMBがイントラモード(intra mode)の場合、及び、予測ブロックがVOPの境界に位置するために、隣接ブロックがない場合、及び、予測ブロックがマクロブロックMBの底辺(ブロック2、3)に位置するため、真下に隣接するブロックの動きベクトルMVを用いることが出来ない場合には、これら隣接ブロックに対応する動きベクトルMVは現在ブロック(current block)の動きベクトルMVを代用する。

【0126】以上が、4:2:0フォーマットにおける、動きベクトル、及び動き補償の計算法であるが、マクロブロック内の色差ブロックの個数の異なる、4:2:2、及び4:4:4フォーマットに対しては適用することが出来ない、以下に4:2:2、及び4:4:4フォーマットのための動きベクトル、及び動き補償の計算方法を示す。

【0127】図13のフローチャートでは、ステップS205においてクロマタイプ(chroma_type)が“01”以外、すなわちNOと判別され、ステップS221で、4:2:2フォーマットである(chroma_type=“10”)か否かの判別が行われる。ステップS221でYES(4:2:2フォーマット)と判別されたときには、ステップS222以降の4:2:2フォーマットにおける色差成分の動きベクトル計算及び動き補償が行われ、NO(4:4:4フォーマット)と判別されたときには、ステップS231以降の4:4:4フォーマットにおける色差成分の動きベクトル計算及び動き補償が行われる。

【0128】4:2:2、4:4:4フォーマットいずれにおいても、輝度ブロックに対するMV、MCの計算法は4:2:0と同じである。

【0129】4:2:2フォーマットにおけるマクロブ

ロックMBの構造は、図3の(b)となる。この時、C_b、C_rの上部のブロック4、5は同じ動きベクトルMVを持つ。下部のブロック6、7についても同様に、同じ動きベクトルMVを持つ。

【0130】まず、マクロブロックMBのモードが図3 4におけるインター(inter)又はインター+q (inter+q)モードの場合は、色差成分の全てのブロックの動きベクトルMVは全て同じ値となり、図15の(a)に示すように、色差成分のx方向の解像度が1/2であるということから、動きベクトルのx成分のみを1/2にする。ここで、MVCのx成分MVC_xはfourth sample resolutionであるため、図17の(b)を適用して、half sample resolutionに変換される。

【0131】次に、マクロブロックMBのモードがインター4v (inter4v)モードの場合は、図15の(b) (c)に示すように、色差成分の上部のブロック5、6の動きベクトルMVは、マクロブロックの上部の2個の輝度ブロック(ブロック0、1)の動きベクトルMV (MV₀、MV₁)から計算される。ただし、図15の(c)に示すように、対応する2個の輝度ブロックのいずれかがオブジェクトの外部となった場合には、オブジェクト内の輝度ブロックのベクトルから色差ブロックのベクトルを計算し、対応する輝度ブロックの両方がオブジェクトの外部となった場合には、動き補償はされない。

【0132】ここで、図15の(b)、(c)のK(0 < K ≤ 2)を、輝度ブロック0、1のうちのオブジェクトの内部にあるブロックの個数とする。K=1の場合、MVC4のx成分はfourth sample resolutionとなるため、図17の(d)を用いて、half sample resolutionに変換される。K=2の場合、MVC4のx成分、y成分はそれぞれeighth sample resolution、fourth sample resolutionとなるため、図17の(c)、(d)を用いて、half sample resolutionに変換される。K=0の場合は、色差ブロック4、6に対応する2個の輝度ブロック全てがオブジェクトの外部となるため、このブロックは符号化されない。色差成分の下部のブロック6、7の動きベクトルMVは、対応する2個の輝度ブロック2、3の動きベクトルMV₂、MV₃から計算される。その計算法は、上部の色差ブロック4、5の場合と同じである。

【0133】4:2:2フォーマットの場合、色差ブロックの解像度は、輝度ブロックと比較して、x方向は解像度が1/2であるが、y方向は同じ解像度であるため、y方向に対しては、近接ブロックの動きベクトルの相関が高い。そのため、色差ブロックの動き補償の方式として、4:2:0のとき用いられた1個の動きベクトルによる動き補償でなく、輝度ブロックの動き補償法であるオーバーラップ動き補償を、4:2:2の色差成分用に改良した方式を用い、予測色差ブロックの上下の

隣接ブロックのみの動きベクトルMVを用いてオーバーラップ動き補償を行うことが出来る。これは、VOL syntaxのobmc_disable = 0の時のみ実行される。

【0134】具体的には、図18の(b)のように上下のブロックを隣接ブロックとし、予測色差ブロック内の各画素値は現在の色差ブロックの動きベクトルMVと1個の隣接ブロックの動きベクトルMVから得られた2個の予測画素値に図19の(a)、(b)のmatrixに示す重みを掛けて足し合わせ、8で割ることより得られる。

例えば、予測色差ブロックの上半分に有る画素に対しては、このブロックの真上に隣接するブロックの動きベクトルMVが使われ、逆に下半分の画素に対しては、真下に隣接するブロックの動きベクトルMVが用いられる。この計算を各画素に対して一般化した式が、以下である。

【0135】すなわち、予測輝度ブロック内の各画素値をp(i, j)とすると、

$$p(i, j) = (q(i, j) \times H-I0(i, j) + r(i, j) \times H-I1(i, j) + 4) // 8$$

ここで、q(i, j)、及びr(i, j)は、参照画像から得られる画素値であり、

$$q(i, j) = q(i + MV_x, j + MV_y),$$

$$r(i, j) = r(i + MV_x, j + MV_y)$$

である。また、ベクトル(MV_x, MV_y)は、現在のブロックの動きベクトルであり、ベクトル(MV'_x, MV'_y)は、上部、又は下部隣接ブロックの動きベクトルである。さらに、行列H-I0(i, j)、及びH-I1(i, j)は、図20の(a)、(b)で定義された行列である。

【0136】ただし、隣接ブロックが符号化されていない場合は、このブロックはMV=0とする。また、隣接ブロックに対応するマクロブロックMBがイントラモード(intra mode)の場合、及び、予測色差ブロックがVOPの境界に位置するために、隣接ブロックがない場合、及び、予測色差ブロックがマクロブロックMBの底辺(図におけるブロック6、7)に位置するため、真下に隣接するブロックのMVを用いることが出来ない場合には、これら隣接ブロックに対応する動きベクトルMVは現ブロックの動きベクトルMVを代用する。

【0137】図13のフローチャートにおいては、ステップS222で4:2:2フォーマットにおける色差ブロックの動きベクトル計算が行われ、ステップS223で上記フラグobmc_disableが0が否かの判別がなされ、obmc_disable = 0 (YES)の場合にはステップS225に進んでオーバーラップ動き補償が行われ、NOの場合には、ステップS224に進んで通常の(オーバーラップ動き補償でない)動き補償が行われる。

【0138】次に、4:4:4フォーマットの場合の色差ブロックの動きベクトル計算及び動き補償動作について説明する。

【0139】4:4:4フォーマットにおけるマクロブ

ロックMBの構造は図3の(c)となる。この時、C_b、C_rの同じ位置にあるブロック4、5、6、7、8、9、10、11は、それぞれ同じ動きベクトルMVを持つ。

【0140】まず、マクロブロックのモードがインター(inter)又はインター+q(intert+q)モードの場合は、色差ブロックの各ブロックの動きベクトルMVは全て同じ値となり、そのベクトルは、輝度ブロックの動きベクトルMVと同じベクトルとなる(図16の(a))。

【0141】マクロブロックのモードがインター4v(inter4v)モードの場合は、各色差ブロックの動きベクトルMVは、その色差ブロックと同じ位置にある輝度ブロックのMVと同じベクトルとなる(図16の(b)、(c))。ただし、色差ブロックがオブジェクトの外部にある場合は、このブロックは符号化されない。(図16の(c))。

【0142】4:4:4フォーマットの場合、色差ブロックの解像度は、輝度ブロックと同じであるため、隣接ブロックの動きベクトルの相関が高い。そのため、輝度ブロックの動き補償法であるオーバーラップ動き補償を、4:4:4の色差ブロックに対して同様に行なうことが出来る。図13のフローチャートにおいては、ステップS231で4:4:4フォーマットにおける色差ブロックの動きベクトル計算が行われ、ステップS232で上記フラグ obmc_disable が0か否かの判別がなされ、obmc_disable = 0 (YES) の場合にはステップS234に進んでオーバーラップ動き補償が行われ、NOの場合には、ステップS233に進んで通常の(オーバーラップ動き補償でない)動き補償が行われる。

【0143】以上が、各マクロブロックに対する動き補償法である。

【0144】再び図5に戻って、上述したような動き補償が行われる動き補償器42からの予測画像データは、加算器としての演算器40にも供給される。また、前方/後方/両方向予測の場合、演算器40には、予測画像データの他、予測画像によって差分化された差分データが逆DCT回路39から送られている。当該演算器40では、この差分データを動き補償器42からの予測画像に対して加算し、これにより局所復号が行われることになる。この局所復号画像は、復号化装置で復号される画像と全く同一の画像であり、上述したように、次の処理画像に対して、前方/後方/両方向予測を行うときに用いる画像としてフレームメモリ41に記憶される。

【0145】また、予測モードがイントラ符号化である場合、演算器40には画像データそのものが逆DCT回路39の出力として送られてくるので、当該演算器40は、この画像データをそのままフレームメモリ41に出力して記憶させる。

【0146】なお、MPEG4においては、MPEG1

及びMPEG2と異なり、Bピクチャ(B-VOP)も参照画像として用いられるため、当該Bピクチャも局所復号され、フレームメモリ41に記憶されるようになされている。但し、現時点においては、Bピクチャが参照画像として用いられるのは階層符号化を行った場合の上位レイヤについてだけである。

【0147】図5に戻り、DCT係数差分化器44では、量子化器35にて量子化されたDCT係数に対し、後述するように、ブロック単位でAC係数/DC係数の予測処理を行う。

【0148】すなわち、DCT係数差分化器44は、イントラ符号化されるマクロブロックの各ブロックに対し、そのDCT係数のうちのDC係数とAC係数をMPEG4 (ISO/IEC 14496-2)のコミッティドラフト (Committee Draft) にて定められた手段に応じて予測を行う。また、DCT係数差分化器44は、この時、各ブロック毎に同じくコミッティドラフトにて定められた手段に応じて各係数の予測に用いるブロックを決定する。

【0149】図21には、DCT係数差分化器44の具体的構成例を示す。

【0150】この図21において、DCT係数差分化器44には、量子化されたDCT係数及び、量子化スケール、形状情報、色差用形状情報、予測モードが入力される。予測モードは、イントラフラグ発生器400に入力される。イントラフラグ発生器400は、予測モードがイントラ符号化を示す予測モード(イントラ符号化モードあるいは画像内予測モード)である場合のみ、イントラフラグを立てる。このイントラフラグは、通常0となっているものであり、イントラ符号化の予測モードのときに1となれる。

【0151】一方、量子化されたDCT係数、すなわち64個のDCT係数は、スイッチ401に供給される。スイッチ401は、イントラフラグが0の場合、被切換端子A側を選択し、またイントラフラグが1の場合、被切換端子B側を選択するようになされている。従って、予測モードがイントラ符号化(画像内予測)モードでない場合、DCT係数は、スイッチ401の被切換端子Aを介して、図5のVLC器36へ供給される。

【0152】一方、予測モードがイントラ符号化モードの場合、DCT係数は、DCT係数予測器402、及び逆量子化器405へ供給される。

【0153】DCT係数予測器402は、スイッチ401より供給されたDCT係数と、DCT係数予測値選択器403より供給されたDCT係数の予測値との差分をとり、その差分値を図5のVLC器36へと供給する。

【0154】また、逆量子化器405では、後のDCT係数の予測に使用されるDC係数及び一部のAC係数を、図5の逆量子化器38と同様の手段で逆量子化し、その結果をDCT係数予測モード蓄積メモリ404に供給する。

【0155】DCT係数予測モード蓄積メモリ404は、前述した通り、DCT係数の予測に用いるDC係数及び一部のAC係数、さらに予測モードを蓄積する。ここで蓄積された情報は、次段のDCT係数予測値選択器404において使用される。

【0156】DCT係数予測値選択器403では、DC係数と、DCT係数予測モード蓄積メモリ404に保存された情報を用いて、DCT係数の予測値及び予測ブロックの位置を出力し、その値をDCT係数予測器402に供給する。

【0157】ここでDCT成分予測値選択器403の動作を、図22を用いて説明する。図22は、DC係数の差分演算を行うブロックXと、当該差分演算を行う上でDCT係数を参照するブロックの位置関係を示している。

【0158】この図22において、ブロックXの左側に位置するブロックをA、上に位置するブロックをC、その左上に位置するブロックをBとする。また、ブロック*

```
if(|FA[0][0]-FB[0][0]|<|FB[0][0]-FC[0][0]|)
    predict from block C
else
```

```
    predict from block A
```

すなわち、この式(1)は、FA[0][0]とFB[0][0]の差の絶対値とFB[0][0]とFC[0][0]の差の絶対値を比較し、FA[0][0]とFB[0][0]の差の絶対値がFB[0][0]とFC[0][0]のよりも小さい場合には、ブロックAをその予測ブロックとしその予測に用い、FB[0][0]とFC[0][0]の差の絶対値がFA[0][0]とFB[0][0]の差の絶対値よりも小さい場合にはブロックCをその予測ブロックとし予測値とすることを表している。

【0161】また、DCT係数は、入力された量子化ステップにより図5の量子化器35と同様の方法により量子化される。これは、DCT係数予測器402において入力されたDCT係数は、DCT係数予測値選択器403に入力された量子化スケールと同じ値により既に量子化されているため、その量子化ステップを揃えるためである。

【0162】また、予測に用いられるDCT係数は、図23により示されるAC係数及び各DC係数である。

【0163】以下、予測値の選択方法をこの図23を用いて説明する。

【0164】図23において、ブロックXのDCT係数の予測ブロックがブロックAであった場合、その予測に用されるAC係数は、ブロックAにおいて図中黒色で示される位置に対応するAC係数となり、またブロックAのDC係数も同様に予測に用いられる。また、ブロックCがその予測ブロックとされた場合、当該ブロックCの図中黒色で示される位置に対応するAC係数及び、当該ブロックCのDC係数がその予測値として用いられる。

* AのDC係数をFA[0][0]、ブロックBのDC係数をFB[0][0]、ブロックCのDC係数をFC[0][0]とする。これらの値は、逆量子化されたDC係数の値であり、量子化されたブロックA、B、CのDC係数に、イントラDCプレシジョン(intra_dc_precision)、量子化スケールを用いて図7の手法により求められたDC量子化ステップの値を積算して求められたものである。ただし、これらのブロックが、VOPの外側に位置していたり、イントラ(intra)符号化されるブロック以外のブロックであった場合、2の(number_of_bits-2)乗の値が使用される。また、図22のA、B、Cの各ブロックについて、これらのブロックがオブジェクトの外側であると判断された場合、これらのDC係数も(number_of_bits-2)乗の値となる。

【0159】ここで、ブロックXのDC係数の差分値を求めるのに使用するブロックは、式(1)により決定される。

【0160】

(1)

【0165】さらに、ブロックAが予測ブロックとして選択された場合、ブロックXにおいては、そのDC係数及び左側の列に並んだAC係数のみがその予測の対象となり、これらのDCT係数のみの差分がとられることになる。また同様に、ブロックCが予測ブロックとして選択された場合、ブロックXにおいては、一番上の行のAC係数及びDC係数がその予測対象となる。

【0166】また、DC係数の予測値は、その予測に選択されたブロックがオブジェクトの外側に位置していたり、インター予測をしていた場合、128の値が出力され、AC係数の場合は、選択されたブロックがオブジェクトの外側に位置したとき、そのAC係数が0として扱われる。

【0167】DCT成分予測値選択器403において上述のようにして求められたDCT係数の予測値は、DCT係数予測器402に供給される。

【0168】DCT係数予測器402では、前述したようにDCT係数の予測値と、それに対応する位置の差分を求め、その値を図5のVLC器36に出力する。

【0169】次に、上述したようなAC/DC予測に用いるブロックについて、色差フォーマット毎に図24を用いて説明する。また、色差ブロックにおいてオブジェクト内のブロックかどうか判別する方法について、図25を用いて説明する。

【0170】図24の(a)は、輝度及び4:4:4フォーマットの色差ブロックにおけるAC/DC予測に用いるブロックを模式的に示したものである。

【0171】すなわち、図24の(a)におけるブロック

クEのAC/DC係数の予測を行う場合、先ず、A, B, Dのブロックを用いて、その予測に用いるブロックを決定する。その後、その決定に従い、ブロックBもしくはブロックDの何れかのブロックが、ブロックEの予測ブロックとして用いられることになる。同様に、ブロックFのAC/DC係数の予測は、B, C, Eのブロックを予測の決定に用い、その後当該決定に従ってC, Eの何れかのブロックから予測を行う。ブロックHのAC/DC係数の予測は、D, E, Gのブロックを予測の決定に用い、その後当該決定に従ってE, Gのブロックの何れかから予測を行う。ブロックIのAC/DC係数の予測は、E, F, Hのブロックを予測の決定に用い、その後当該決定に従ってF, Hのブロックの何れかから予測を行う。

【0172】ここで、AC/DC予測を行う際には、各ブロックがオブジェクトの境界内に存在するか否かを判定する必要がある。

【0173】輝度ブロックの場合は、図5の形状情報復号器5.2より供給される形状情報を用いてこれを決定する。すなわちオブジェクトの内側かどうかを判定するのは、その同位置に位置する形状情報を参照し、同位置の形状情報の8×8画素のブロック内に、形状情報が1の値となる画素を持つ画素値が一つ以上存在する場合(すなわちオブジェクト内の画素が1つ以上存在する場合)、そのブロックをオブジェクト内の画素として見なすことができる。

【0174】4:4:4フォーマットにおける色差ブロックの場合は、図25に示すように、図5の色差用形状情報作成器5.3より供給される色差用の形状情報(ただし、4:4:4フォーマットの場合は、輝度に用いられる形状情報と等しい)から各ブロック内に値が1となる画素値を持つ画素、すなわちオブジェクトの内側であることを示す画素が1つ以上存在するか否かで判別することができる。

【0175】また、これ以外の方法としては、同位置に存在する輝度のブロックがオブジェクト内か否かにより判別しても良い。例えば図3の(c)の場合、ブロック4, 5がオブジェクトの内側であるか否かは、ブロック0より判定することができ、同様にブロック6, 7の場合はブロック2より、ブロック8, 9の場合はブロック1より、ブロック10, 11の場合はブロック3より判定することも可能である。

【0176】それらどちらの判断方法を使用した場合でも、色差ブロックがオブジェクトの内側に位置するかどうかの判定結果は同じになるため、その符号化/復号化装置の仕様により適宜選択することができる。

【0177】図24の(b)は、4:2:2フォーマットの色差ブロックにおけるAC/DC予測に用いるブロックを模式的に示したものである。

【0178】すなわち、図24の(b)におけるブロック

クDのAC/DC係数の予測を行う場合、先ずA, B, Cのブロックを用いて、その予測に用いるブロックを決定する。その後当該決定に従ってBもしくはCのブロックの何れかがブロックDの予測ブロックとして用いられることになる。同様にブロックGのAC/DC係数の予測は、C, D, Eのブロックを予測の決定に用い、その決定に従ってD, Eのブロックの何れかから予測を行う。

【0179】この4:2:2フォーマットにおいても、上記4:4:4フォーマットの色差ブロックの場合と同様に、AC/DC予測を行う上で各ブロックがオブジェクトの境界内に存在するか否かを判定する必要がある。

【0180】すなわち当該4:2:2フォーマットにおける色差ブロックの場合は、図25に示すように、図5の色差用形状情報作成器5.3より供給される色差用の形状情報から、各ブロック内に値が1となっている画素値を持つ画素、すなわちオブジェクトの内側であることを示す画素が存在するか否かで判別することができる。

【0181】また、これ以外の方法としては、判定を行う色差ブロックと対応する位置に存在する輝度のブロックがオブジェクト内か否かにより判別しても良い。例えば図3の(b)の場合、ブロック4, 5がオブジェクトの内側かどうかは、ブロック0, 1の何れか、もしくは両方がオブジェクトの内側のブロックであるとき、当該ブロック4, 5もオブジェクトの内側のブロックであると判断することができ、同様にブロック6, 7の場合は、ブロック2, 3がオブジェクトの内側かどうかで判断することができる。

【0182】どちらの判断方法を使用した場合でも色差ブロックがオブジェクトの内側に位置するかどうかの判定結果は同じになるため、その符号化/復号化装置の仕様により適宜選択することができる。

【0183】図24の(c)は、4:2:0フォーマットの色差ブロックにおけるAC/DC予測に用いるブロックを模式的に示したものである。

【0184】すなわち、図24の(c)におけるブロックDのAC/DC係数の予測を行う場合、先ずA, B, Cのブロックを用いて、その予測に用いるブロックを決定する。その後その決定に従ってBもしくはCのブロックの何れかが、Dのブロックの予測ブロックとして用いられることになる。

【0185】4:2:0フォーマットにおいても、上記4:4:4フォーマットや4:2:2フォーマットの色差ブロックの場合と同様に、AC/DC予測を行う上で各ブロックがオブジェクトの境界内に存在するか否かを判定する必要がある。

【0186】すなわち4:2:0フォーマットにおける色差ブロックの場合、図25に示すように、図5の色差用形状情報作成器5.3より供給される色差用の形状情報から、各ブロック内に値が1となる画素値を持つ画素、

すなわちオブジェクトの内側であることを示す画素が存在するか否かで判別することができる。

【0187】またこれ以外の方法としては、判定を行う色差ブロックと対応する位置に存在する輝度のブロックがオブジェクト内か否かにより判別しても良い。例えば図3の(a)の場合、ブロック4, 5がオブジェクトの内側かどうかは、ブロック0, 1, 2, 3の何れか、もしくは両方がオブジェクトの内側のブロックであるとき、これらブロック4, 5もオブジェクトの内側のブロックであると判断することができる。

【0188】どちらの判断方法を使用した場合でも、色差ブロックがオブジェクトの内側に位置するかどうかの判定結果は同じになるため、その符号化/復号化装置の仕様により適宜選択することができる。

【0189】図5に戻って、VLC器36では、図42における場合と同様に、量子化係数、量子化ステップ、動きベクトル、および予測モードが供給される他、サイズデータFSZ_BおよびオフセットデータFPOS_Bも供給される。したがって、VLC器36では、これらのデータすべてを可変長符号化する。

【0190】また、VLC器36では、図42で説明したように、I, P, Bピクチャ(I-VOP, P-VOP, B-VOP)のマクロブロックについて、スキップマクロブロックとするかどうかを決定し、その決定結果を示すフラグCOD, MODBを設定する。このフラグCOD, MODBは、やはり可変長符号化されて伝送される。また、各ブロック毎にそのDCT係数の符号化を行うが、そのブロックがオブジェクトの外側のブロックである場合、DCT係数の符号化は行われない。

【0191】各ブロックがオブジェクトの外側もしくは内側に位置するかは、輝度ブロックの場合は形状情報復号器52より判断することができ、色差ブロックの場合はこの輝度ブロックの判定結果もしくは、色差用形状情報作成器53より供給される色差用の形状情報より判断できる。オブジェクトの内側かどうかの判定法は、輝度、色差の場合共に、前述したDCT係数差分化器44におけるブロックの判定法と同様の方法で判定することができる。

【0192】次に、図26には、図1のエンコーダから出力されるビットストリームを復号するデコーダの一実施の形態の構成例を示している。

【0193】このデコーダには、図1のエンコーダから伝送路5または記録媒体6を介して提供されるビットストリーム(bitstream)が供給される。すなわち、図1のエンコーダから出力され、伝送路5を介して伝送されてくるビットストリームは図示せぬ受信装置で受信され、また、記録媒体6に記録されたビットストリームは図示せぬ再生装置で再生されて、逆多重化部71に供給される。

【0194】逆多重化部71では、入力されたビットス

トリーム(後述するVS(Video Stream))を、ビデオオブジェクト(Video Object)ごとのビットストリームVO#1, VO#2, ..., VO#Nに分離し、それぞれ対応するVOP復号部721~72Nに供給する。

【0195】VOP復号部721~72Nのうち、VOP復号部72nでは、逆多重化部71から供給されたビットストリームより、ビデオオブジェクト(Video Object)を構成するVOP(画像データ)、サイズデータ(VOP_size)、およびオフセットデータ(VOP_offset)を復号し、画像再構成部73に供給する。

【0196】画像再構成部73では、VOP復号部721乃至72Nからの出力に基づいて、元の画像を再構成する。この再構成された画像信号は、例えばモニタ74に供給される。これにより、当該モニタ72には再構成された画像が表示される。

【0197】次に、図27は、VOP復号部72nの構成例を示す。なお、図27中で、図43のデコーダにおける場合と対応する各構成要素については、同一の指示符号を付してある。すなわち、VOP復号部72nは、基本的に図43のデコーダと同様に構成されている。

【0198】この図27において、逆多重化部72nから供給されたビットストリームは、バッファ101に供給されて一時記憶される。

【0199】IVLC器102は、後段におけるブロックの処理状態に対応して、上記バッファ101からビットストリームを適宜読み出し、そのビットストリームを可変長復号することで、量子化係数、動きベクトル、予測モード、量子化ステップ、サイズデータFSZ_B、オフセットデータFPOS_B、形状復号化情報およびフラグCODなどを分離する。量子化係数および量子化ステップは、逆量子化器103に供給され、動きベクトルおよび予測モードは、動き補償器107、逆量子化器103、DCT係数逆差分化器111に供給される。また、サイズデータFSZ_BおよびオフセットデータFPOS_Bは、動き補償器107、図26の画像再構成部73に供給される。形状復号化情報は、形状復号化器110に供給される。

【0200】逆量子化器103、IDCT器104、演算器105、フレームメモリ106、形状復号化器100、色差用形状情報作成器109、パディング処理器108、動き補償器107では、図1のVOP符号化部3nを構成する図5の逆量子化器38、IDCT器39、演算器40、フレームメモリ41、形状情報復号器52、色差用形状情報作成器53、パディング処理器51または動き補償器42における場合とそれぞれ同様の処理が行われる。

【0201】形状情報復号器110では、IVLC器102より供給された符号化された形状情報を復号して復元する。形状情報復号器110は、図5の形状情報復号器52と同様に動作し、その出力はDCT係数逆差分化

器111、パディング処理器108、及び色差用形状情報作成器109に供給される。

【0202】色差用形状情報作成器109は、図5の色差用形状情報作成器53と同様に動作し、色差のフォーマットに合わせて、形状情報復号器110からの出力を変換してパディング処理器108及びDCT係数逆差分化器111に供給する。

【0203】逆量子化器103では、IVLC器102から供給された量子化係数（量子化スケール）を元に、同じくIVLC器102から供給された量子化されたDCT係数を逆量子化する。当該逆量子化器103は、図5の逆量子化器38と同様に動作し、その出力はDCT係数逆差分化器111に供給される。

【0204】DCT係数逆差分化器111は、図5のDCT係数差分化器44と同様にして、該当ブロックの予測に使用されたブロックを選択し、逆量子化器103より供給されたDCT係数と加算することで、DCT係数のAC係数及びDC係数を復元する。このように復元されたDCT係数は、IDCT器104に送られる。

【0205】また、DCT係数逆差分化器111では、図5のDCT係数差分化器44の場合と同様に、形状情報復号器110から供給された形状情報、及び色差用形状情報作成器109より供給された色差用形状情報を用いて、各ブロックにオブジェクトの内側かあるいは外側かの判定を行う。この判定方法についても、図5でのDCT係数差分化器44と同様の方法が使用される。

【0206】IDCT器104は、図5のIDCT器39と同様に動作し、DCT係数逆差分化器111からのDCT係数をIDCT処理し、得られたデータを演算器105に供給する。

【0207】演算器105は、図5の演算器40と同様に動作し、フレーム間予測が行われている場合には動き補償器107からの出力とIDCT器104の出力を1画素単位で加算する。フレーム内予測（イントラ符号化）が行われている場合には、特に動作しない。演算器105の出力は、パディング処理器1108に供給されるとともに、テクスチャ情報として出力される。

【0208】パディング処理器108は、図5のパディング処理器51と同様に動作し、形状情報復号器110及び色差用形状情報作成器109から出力された形状情報を元に、演算器105からの出力画像に対してパディング処理を行う。このパディング処理器108の出力は、フレームメモリ106に蓄積される。

【0209】フレームメモリ106に蓄積されているデータは、動き補償器107により適宜呼び出され、演算器105に出力される。これらフレームメモリ106及び動き補償器107の動作は、図5に示すフレームメモリ41及び動き補償器42の動作と同様である。

【0210】この図27に示したように、VOP復号部72nではVOPが復号され、当該復号されたVOPが

図26の画像再構成部73に供給される。

【0211】次に、図1のエンコーダが出力する符号化ビットストリームのシンタックスについて、例えば、MPEG4規格のビジュアルコミッティドラフト（Visual Committee Draft）の記載を例に説明する。なお、これ以降に説明する各シンタックスは、MPEG4のビジュアルコミッティドラフトにて規定されており、既知のものであるためそれらの図示は省略するが、本発明において特に必要と認めるシンタックスについては図示する。

【0212】MPEG4において、ビジュアルオブジェクト（visual object）については、2次元の動画像データだけではなく、2次元静止画像、フェイスオブジェクト（face object）等を符号化／復号化することが可能である。

【0213】そのため、MPEG4のビジュアルオブジェクト（visual Object）については、先ず、ビジュアルオブジェクトシーケンス（Visual Object Sequence）を伝送する。このビジュアルオブジェクトシーケンス内では、複数のビジュアルオブジェクト（Visual Object）を伝送することができ、複数のビジュアルオブジェクトによりビジュアルオブジェクトシーケンスを構成することが可能である。

【0214】次に、図示しないビジュアルオブジェクト（Visual Object）のシンタックスでは、当該シンタックス以降に続くオブジェクトのタイプ（現在のところ動画像であるビデオオブジェクト（Video Object）、静止画像であるスチルテクスチャオブジェクト（Still Texture Object）、2次元のメッシュを示すメッシュオブジェクト（Mesh Object）、顔形状を示すフェイスオブジェクト（Face Object）の4つが定義されている）等を示すことが可能であり、ビジュアルオブジェクト（Visual Object）に続いて、ビデオオブジェクト（Video Object）、スチルテクスチャオブジェクト（Still texture Object）、メッシュオブジェクト（Mesh Object）、フェイスオブジェクト（Face Object）の何れかのシンタックスが続く。

【0215】また、図示しないユーザデータ（user_data）のシンタックスでは、ビジュアルオブジェクトシーケンス（Visual Object Sequence）、ビジュアルオブジェクト（Visual Object）、ビデオオブジェクト（Video Object）、GOVレイヤ等で使用し、符号化時において、復号時等に使用するデータを復号側で定義して伝送することが可能である。

【0216】ここで、図示しないビデオオブジェクト（VO）のシンタックスにおいて、当該ビデオオブジェクトは、1以上のVOL（Video Object Layer Class）から構成される。なお、画像を階層化しないときは一のVOLで構成され、画像を階層化する場合には、その階層数だけのVOLで構成される。

【0217】続いて、図示しないVOL（video Object

Layer)のシンタックスにおいて、当該VOLは、上述したようなスケーラビリティのためのクラスであり、ビデオオブジェクトレイヤID (video_object_layer_id) で示される番号によって識別される。すなわち、例えば、下位レイヤのVOLについてのビデオオブジェクトレイヤID (video_object_layer_id) は0とされ、また、例えば、上位レイヤのVOLについてのビデオオブジェクトレイヤID (video_object_layer_id) は1とされる。なお、上述したように、スケーラブルのレイヤの数は2に限られることなく、1や3以上を含む任意の数とすることができます。

【0218】また、VOLは大きく2つのシンタックスから構成されており、ビデオオブジェクトレイヤスタートコード (video_object_layer_start_code) からなる部分と、ショートビデオスタートマーカ (short_video_start_marker) より始る部分の2つにより構成されている。

【0219】ここで、VOL (video Object Layer) のシンタックスにおいて、ビデオオブジェクトレイヤスタートコード (video_object_layer_start_code) に続く部分から構成されるシンタックスについて、図2-8を用いて説明する。なお、図2-8には、当該VOLのシンタックスの前半部分のみを示し、後半部分については省略している。

【0220】なお、この図2-8や、後述する図3-0、図3-1、図4-0、図4-1のようなシンタックスを示す図面中で「*」を付した部分は、本実施の形態による変更箇所等の注目部分を表している。

【0221】図2-8に示すシンタックスにおいて、クロマフォーマット (chroma_format) は、色差フォーマットの種類を表し、このクロマフォーマットは図2-9に示す通りに定義される。すなわち、クロマフォーマット (chroma_format) が「01」の場合、符号化/復号化される画像は4:2:0フォーマットであり、また「10」の場合は4:2:2フォーマット、「11」の場合は4:4:4フォーマットとなる。

【0222】また、各VOLについて、それが画像全体であるのか、画像の一部であるのかは、ビデオオブジェクトレイヤシェイプ (video_object_layer_shape) で識別される。このビデオオブジェクトレイヤシェイプ (video_object_layer_shape) は、VOLの形状を示すフラグであり、例えば、以下のように設定される。

【0223】すなわち、VOLの形状が長方形であるとき、当該ビデオオブジェクトレイヤシェイプ (video_object_layer_shape) は、例えば「00」とされる。また、VOLが、ハードキー (0または1のうちの何れか)

hor_sampling_factor_n / hor_sampling_factor_m

で与えられる。

【0228】さらに、バーチカルサンプリングファクタn (ver_sampling_factor_n) とバーチカルサンプリ

*一方の値をとる2値 (バイナリ) の信号) によって抜き出される領域の形状をしているとき、ビデオオブジェクトレイヤシェイプ (video_object_layer_shape) は、例えば「01」とされる。さらに、VOLが、ソフトキー (0乃至1の範囲の連続した値 (グレイスケール) をとることが可能な信号) によって抜き出される領域の形状をしているとき (ソフトキーを用いて合成されるものであるとき)、当該ビデオオブジェクトレイヤシェイプ (video_object_layer_shape) は、例えば「10」とされる。

【0224】ここで、ビデオオブジェクトレイヤシェイプ (video_object_layer_shape) が「00」とされるのは、VOLの形状が長方形であり、かつ、そのVOLの絶対座標系における位置および大きさが、時間とともに変化しない場合、すなわち、一定の場合である。なお、この場合、その大きさ (横の長さと縦の長さ) は、ビデオオブジェクトレイヤワイド (video_object_layer_width) とビデオオブジェクトレイヤハイ (video_object_layer_height) によって示される。ビデオオブジェクトレイヤワイド (video_object_layer_width) およびビデオオブジェクトレイヤハイ (video_object_layer_height) は、いずれも10ビットの固定長のフラグであり、ビデオオブジェクトレイヤシェイプ (video_object_layer_shape) が「0.0」の場合には、最初に一度だけ、伝送される。これはビデオオブジェクトレイヤシェイプ (video_object_layer_shape) が「00」の場合、上述したように、VOLの絶対座標系における大きさが一定であるからである。

【0225】また、VOLが、下位レイヤまたは上位レイヤのうちの何れかであるかは、1ビットのフラグであるスケーラビリティ (scalability) によって示される。VOLが下位レイヤの場合、スケーラビリティ (scalability) は例えば1とされ、それ以外の場合は例えば0とされる。

【0226】さらに、VOLが、自身以外のVOLにおける画像を参照画像として用いる場合、その参照画像が属するVOLは、リファレンスレイヤID (ref_layer_id) で表される。なお、リファレンスレイヤID (ref_layer_id) は、上位レイヤについてのみ伝送される。

【0227】また、ホリゾンタルサンプリングファクタn (hor_sampling_factor_n) とホリゾンタルサンプリングファクタm (hor_sampling_factor_m) は、下位レイヤのVOPの水平方向の長さに対応する値と、上位レイヤのVOPの水平方向の長さに対応する値をそれぞれ示す。従って、下位レイヤに対する上位レイヤの水平方向の長さ (水平方向の解像度の倍率) は、下記式(2)

(2)

グファクタm (ver_sampling_factor_m) は、下位レイヤのVOPの垂直方向の長さに対応する値と、上位レイヤのVOPの垂直方向の長さに対応する値をそれぞれ示

す。従って、下位レイヤに対する上位レイヤの垂直方向* $\text{ver_sampling_factor_n} / \text{ver_sampling_factor_m}$ の長さ（垂直方向の解像度の倍率）は、下記式（3）

で与えられる。

【0229】VOLにおいて、コンプレクシティエスティメイションディセーブル (complexity_estimation_disable) が0の場合、当該シンタックスのディファイン VOPエスティメイションヘッダ (define_VOP_estimation_header) 以下が符号化/復号化される。

【0230】また、VOLのビットストリームの先頭にビデオオブジェクトレイヤスタートコード (video_object_layer_start_code) ではなく、ショートビデオスタートマーカ (short_video_start_marker) が読み込まれた場合、当該シンタックスのelse行以下のシンタックスがVOLのビットストリームの復号に際し使用される。

【0231】VOLは、単数もしくは複数のVOP、ビデオプレーンウイズショートヘッダ (video_plane_with_short_header) もしくはGOVで構成される。

【0232】GOV層は、ビットストリームの先頭だけでなく、符号化ビットストリームの任意の位置に挿入することができるよう、VOL層とVOP層との間に規定されている。これにより、あるVOL#0が、OP#0, VOP#1, ..., VOP#n, VOP#(n+1), ..., VOP#mといったVOPのシーケンスで構成される場合において、GOV層は、その先頭のVOP#0の直前だけでなく、VOP#(n+1)の直前にも挿入することができる。従って、エンコーダにおいて、GOV層は、例えば符号化ストリームの中のランダムアクセスさせたい位置に挿入することができ、従って、GOV層を挿入することで、あるVOLを構成するVOPの一連のシーケンスは、GOV層によって複数のグループ (GOV) に分けられて符号化されることになる。

【0233】図示しないGOV層のシンタックスにおいて、当該GOV層は、グループスタートコード (group_start_code)、タイムコード (time_code)、クローズドGOP (closed_gop)、ブローカンリンク (broken_link)、ネクストスタートコード (next_start_code()) が順次配置されて構成される。

【0234】次に、図示しないVOP (Video Object Plane Class) のシンタックスにおいて、VOPの大きさ（横と縦の長さ）は、例えば、13ビット固定長のVOPワイド (VOP_width) とVOPハイ (VOP_height) で表される。また、VOPの絶対座標系における位置は、例えば13ビット固定長のVOPホリゾンタルスペシャルMCリファレンス (VOP_horizontal_spatial_mc_ref) とVOPバーチカルMCリファレンス (VOP_vertical_mc_ref) で表される。なお、VOPワイド (VOP_width) 又はVOPハイ (VOP_height) は、VOPの水平方向又は垂直方向の長さをそれぞれ表し、これらは上述のサイズデータFSZ_BやFSZ_Eに相当する。また、VOPホリゾンタルスペシャルMCリファ

レンス (VOP_horizontal_spatial_mc_ref) またはVOPバーチカルMCリファレンス (VOP_vertical_mc_ref) は、VOPの水平方向または垂直方向の座標 (x座標またはy座標) をそれぞれ表し、これらは、上述のオフセットデータFPOS_BやFPOS_Eに相当する。

【0235】VOPワイド (VOP_width)、VOPハイ (VOP_height)、VOPホリゾンタルスペシャルMCリファレンス (VOP_horizontal_spatial_mc_ref)、VOPバーチカルMCリファレンス (VOP_vertical_mc_ref) は、ビデオオブジェクトレイヤシェイプ (video_object_layer_shape) が「00」以外の場合にのみ伝送される。すなわち、ビデオオブジェクトレイヤシェイプ (video_object_layer_shape) が「00」の場合、上述したように、VOPの大きさおよび位置はいずれも一定であるから、それらVOPワイド (VOP_width)、VOPハイ (VOP_height)、VOPホリゾンタルスペシャルMCリファレンス (VOP_horizontal_spatial_mc_ref)、VOPバーチカルMCリファレンス (VOP_vertical_mc_ref) は伝送する必要がない。この場合、受信側では、VOPは、その左上の頂点が、例えば、絶対座標系の原点に一致するように配置され、また、その大きさは、上述のビデオオブジェクトレイヤスタートコード (video_object_layer_start_code) に続く部分から構成されるシンタックスのところで説明したビデオオブジェクトレイヤワイド (video_object_layer_width) およびビデオオブジェクトレイヤハイ (video_object_layer_height) から認識される。

【0236】次に、リファレンスセレクトコード (ref_select_code) は、参照画像として用いる画像を表すもので、VOPのシンタックスにおいて規定されている。

【0237】当該シンタックスのリードVOPコンプレクシティエスティメイションヘッダ (read_VOP_complexity_estimation_header) 以下は、VOLにおいてコンプレクシティエスティメイションディセーブル (complexity_estimation_disable) の値が「0」と示された場合のみVOPより、読み込まれる。

【0238】また、ビデオパケットヘッダ (video_packet_header) は、VOLにおいて、エラーレジッシュエンティディセーブル (error_resilient_disable) が「0」を示す時にのみ使用可能であり、符号化側でその使用を自由に使用することができVOPより読み込むことが可能である。

【0239】ここで、前記VOLがショートビデオスタートマーカ (short_video_start_marker) より始る場合において使用されるビデオプレーンウイズショートヘッダ (video_plane_with_short_header) について説明する。

【0240】ビデオプレーンウイズショートヘッダ (video_plane_with_short_header) は、前述した通り VOP がショートビデオスタートマーカ (short_video_start_marker) により開始する場合にのみ使用される。

【0241】このショートビデオスタートマーカ (short_video_start_marker) は、フラグ群及び複数の GOB レイヤ (gob_layer) により構成される。

【0242】GOB レイヤ (gob_layer) は、複数のマクロブロックをひとまとまりのグループとして符号化したもので、GOB レイヤ (gob_layer) 内のマクロブロックの個数は符号化を行っている画像の画枠により一意に定められる。

【0243】ビデオオブジェクトプレーン (video_object_plane) では、画像のテクスチャ情報、形状符号化情報を符号化するために、モーションシェイプテクスチャ (motion_shape_texture) が読み込まれ、この中でマクロブロック等の符号化が行われる。図示しないモーションシェイプテクスチャ (motion_shape_texture) のシンタックスは、大きく 2 つの部分から構成され、データパーティショニングモーションシェイプテクスチャ (data_partitioning_motion_shape_texture) と、コンパインドモーションシェイプテクスチャ (combined_motion_shape_texture) に分けられる。データパーティショニングウモーションシェイプテクスチャ (data_partitioning_motion_shape_texture) は、VOL で示される 1 ビットのフラグ (data_partitioning) が 1 の場合であり、テクスチャ情報が伝送される場合に使われる。

【0244】コンパインドモーションシェイプテクスチャ (combined_motion_shape_texture) は、フラグ (data_partitioning) が 0 の場合もしくは、形状情報をのみを伝送する場合に用いられる。このようにコンパインドモーションシェイプテクスチャ (combined_motion_shape_texture) は、一つもしくは複数のマクロブロックにより構成されている。

【0245】図 30 及び図 31 には、マクロブロックのシンタックスの一部 (前半部分) を示す。このマクロブロックのシンタックスは大きく 3 つの部分から構成されており、I-VOP、P-VOP におけるマクロブロックのシンタックスを示す部分と、B-VOP のマクロブロックのシンタックスを示す部分、またグレイスケールシェイプ (GrayScale shape) におけるマクロブロックを示す部分の 3 個所から構成される。

【0246】I-VOP、P-VOP におけるマクロブロックのシンタックスを示す部分、及び B-VOP のマクロブロックのシンタックスを示す部分については、シェイプのブロックを符号化する部分 MB バイナリシェイプコーディング (mb_binary_shape_coding)、マクロブロックの符号化状態を示すフラグ群、及び動きベクトルの符号化部モーションベクター (motion_vector)、各ブロックの符号化部ブロック (block) から構成され

る。

【0247】また、マクロブロックのグレイスケール (GrayScale) の情報を表す部分は、マクロブロック内のグレイスケール (Gray-Scale) 状態を表すフラグ群及びグレイスケール (Gray-Scale) を構成するブロックの符号化部アルファブロック (alpha_block) から構成される。

【0248】当該マクロブロックのシンタックスでは、I-VOP、P-VOP で使用されるものと B-VOP で使用されるものの大きく 2 つのシンタックスより構成される。ここで、I-VOP、P-VOP の時に読み込まれるフラグ (mcBpc) は、マクロブロックのタイプ及び、マクロブロックの色差ブロックのコーディングパターンを示す VLC である。MPEG 4 のビジュアルコミッティドラフト (Visual Committee Draft) で使用されているフラグ (mcBpc) の VLC テーブルを、図 32 に示す。

【0249】この図 32 に示す VLC テーブルは、I-VOP のフラグ (mcBpc) に対応する表であり、図 33 には P-VOP に対する VLC テーブルを示す。

【0250】これら図 32、図 33 の VLC テーブルに示すように、フラグ (mcBpc) はその値により独立した 2 つのパラメータを指す。一つはマクロブロックタイプ (MBtype) であり、他方はフラグ (cbpc) である。フラグ (mcBpc) は図 32、図 33 のテーブルをもとに、符号化/復号される。図 32、図 33 のテーブルを用いて、フラグ (mcBpc) より、マクロブロックタイプ (MBtype) 及び、フラグ (cbpc) が判断される。マクロブロックタイプ (MBtype) はマクロブロックの符号化タイプであり、フラグ (cbpc) は色差信号の符号化の状態を表す 2 ビットのフラグである。マクロブロックタイプ (MBtype) は、整数により記述されているが、各整数は、図 34 のようにマクロブロックの符号化タイプとして定義される。また、同じ番号でもその意味は VOP のタイプにより異なることもある。

【0251】ここで、本実施の形態の方法を明確にするために MPEG 4 のビジュアルコミッティドラフト (Visual Committee Draft) におけるマクロブロックレイヤのフラグ (cbpc) の動作について説明する。フラグ (cbpc) は、図 32 に示されるように 2 ビットで示され、前述したようにそれぞれの色差ブロックの符号化の状態を表す。

【0252】先ず、4:2:0 フォーマットにおけるフラグ (cbpc) の動作を説明する。

【0253】当該フラグ (cbpc) は 2 ビットで表され、各ビットはそれぞれ対応した色差ブロックの符号化状態を表す。すなわち、フラグ (cbpc) のうちの先頭の 1 ビットは図 3 の (a) に示したブロック 4 の符号化状態を示し、後ろの 1 ビットは図 3 の (a) のブロック 5 の符号化状態を示す。このビットの値が 1 の時、その対応す

るブロックは符号化／復号化されるブロックとして判断され、後にこのフラグをもとにブロックのDCT係数が¹⁰出力／入力される。またこのビットが0の場合、対応するブロックのDCT係数は何も存在しないと判断される。

【0254】このようなことから、例えばフラグ(cbpc)が「11」の場合、図3の(a)の対応するブロック4, 5のそれぞれにDCT係数が存在していると判断される。また「10」の場合にはブロック4にのみDCT係数が存在する、「01」の場合には5のブロックのみ、「00」の場合にはブロック4, 5にはDCT係数が存在しないと判断される。

【0255】また、本実施の形態では、他の色差フォーマットにおいてはフラグ(cbpc)を以下のように定義する。

【0256】クロマタイプ(croma_type)が「01」すなわち4:2:0フォーマットを示す場合、フラグ(cbpc)は従来と同じ定義、すなわち4:2:0フォーマットのマクロブロックの構成図を表す図3の(a)において、ブロック4, 5の状態を表すフラグであると判断する。

【0257】一方、クロマタイプ(croma_type)が「10」もしくは「11」の場合、すなわち4:2:2フォーマットもしくは4:4:4フォーマットの場合、フラグ(cbpc)はシンタックスに従い2ビットの値が読み込まれる。ここでこのフラグのセマンティックスは4:2:0フォーマットの場合と異なる。この時、先頭の1ビットは4:2:2フォーマットの場合、図3の(b)における色差Cbのブロック4, 6の何れかもしくは両方がDCT係数を持つか否かを表すフラグとして、また、4:4:4フォーマットの場合、図3の(b)における色差Cbのブロック4, 6, 8, 10の一つ以上のブロックがDCT係数を持つことを示すフラグと判断される。

【0258】残り1ビットは、同様に4:2:2フォーマットの場合、図3の(b)の色差Crのブロック5, 7がDCT係数を持つか否かを表すフラグとして、また、4:4:4フォーマットの場合、図3の(c)の色差Cbのブロック5, 7, 9, 11がDCT係数と持つか否かを示すフラグとして判断される。

【0259】フラグ(cbpc)が「00」を示す場合、4:2:2フォーマット、4:4:4フォーマットどちらの場合においてもどの色差ブロックもDCT係数を持たないものと判断される。

【0260】ここでフラグ(cbpc)が「10」の場合もしくは「11」の場合、色差Cbのブロックにおいて、DCT係数を持つブロックが存在すると判断される。このとき色差Cbのブロックの符号化状態を示すフラグ(cbpcb)が読み込まれる。

【0261】4:2:2フォーマットの場合、フラグ

(cbpcb)は1ビットもしくは2ビットであり、4:4:4フォーマットの場合、フラグ(cbpcb)は1から4ビットとなる。

【0262】フラグ(cbpcb)のビット数は、そのオブジェクトの内側のブロック数により判断される。例えば、4:2:2フォーマットの場合、図3の(b)のブロック4, 5のうち、どちらか1つのブロックのみがオブジェクトの内側であったとき、このマクロブロックでのフラグ(cbpcb)のビット数は1ビットとなり、ブロック4, 5の二つのブロックともオブジェクトの内側であるとき、2ビットのフラグが²⁰出力される。

【0263】同様に、4:4:4フォーマットの場合には、図3の(c)の何れか1つのマクロブロックのみがオブジェクトの内側であると、フラグ(cbpcb)が1ビット、2個オブジェクト内のブロックが存在する場合には2ビット、3個の場合は3ビット、すべてのマクロブロックがオブジェクトの内側である場合には4ビットが³⁰出力される。

【0264】これらフラグ(cbpcb)の各ビットは色差Cbの各ブロックに対し、DCT係数を伝送するマクロブロックか否かを出力することを示すフラグとなる。またこの時、オブジェクトの外側のブロックに対しては、輝度の場合と同様に符号化されるテクスチャのデータが存在しないため、符号化データは何も出力されず、また当然、フラグ(cbpcb)のビットにもその状態を表すフラグは存在しない。

【0265】³⁰このようにフラグ(cbpcb)は、オブジェクト内のブロックに対して、図3に示す順序にて各1ビットづつ割り当てられて、またそのフラグが1の場合、該当ブロック内にDCT係数が存在し、0の場合には存在しないことを示す。

【0266】図35及び図36には上記フラグの定義の手順を示す。なお、このフローチャートでは、図37に示すような色差のブロック番号を使用している。

【0267】ステップS1ではフラグ(cbpc)の読み込みを開始し、ステップS2ではクロマタイプ(croma_type)が「01」か否かの判断を行う。

【0268】当該ステップS2にてクロマタイプ(croma_type)が「01」であると判断されたとき、すなわちクロマタイプ(croma_type)が4:2:0フォーマットを示す場合には、ステップS3にてフラグ(cbpc)は従来同様に定義する。つまり、4:2:0フォーマットのマクロブロックの構成図を表す前記図3の(a)において、ブロック4, 5の状態を表すフラグであると判断する。

【0269】一方、ステップS2において、クロマタイプ(croma_type)が「01」でないと判断したとき、すなわちクロマタイプ(croma_type)が「10」もしくは「11」であると判断(4:2:2フォーマットもしくは4:4:4フォーマットであると判断)した場合は、

ステップS4に進む。

【0270】ステップS4では、フラグ(cbpc)が「01」又は「11」か否かを判断する。このステップS4において、フラグ(cbpc)が「01」又は「11」であると判断した場合はステップS5に進み、そうでないと判断した場合はステップS6に進む。

【0271】ステップS6では、フラグ(cbpc)が「10」又は「11」か否かを判断する。このステップS6において、フラグ(cbpc)が「10」又は「11」であると判断した場合はステップS7に進み、そうでないと判断した場合はステップS8に進んで処理を終了する。

【0272】ステップS5とステップS7では、クロマタイプ(croma_type)が「10」か否かの判断を行い、何れの場合も図36のステップS9に進む。

【0273】図36のステップS9では、オブジェクト内のブロックの個数をカウントし、次のステップS10では、オブジェクト内のブロックの個数分のビットを読み込んでバッファに格納する。

【0274】次のステップS11では、図37に示したブロックc1がオブジェクト内であるか否かを判断する。このステップS11において、ブロックc1がオブジェクト内であると判断した場合はステップS12に進み、そうでないと判断した場合はステップS13に進む。

【0275】ステップS12では、バッファより1ビットを取り出し、ブロックc1の符号化パターンの判断に使用する。

【0276】ステップS13では、図37に示したブロックc2がオブジェクト内であるか否かを判断する。このステップS13において、ブロックc2がオブジェクト内であると判断した場合はステップS14に進み、そうでないと判断した場合はステップS15に進む。

【0277】ステップS14では、バッファより1ビットを取り出し、ブロックc2の符号化パターンの判断に使用する。

【0278】ステップS15では、4:2:2フォーマットであるか否かの判断を行う。このステップS15において、4:2:2フォーマットであると判断した場合はステップS16に進み、そうでないと判断した場合はステップS17に進む。

【0279】ステップS16では、図35の処理に戻る。

【0280】ステップS17では、図37に示したブロックc3がオブジェクト内であるか否かを判断する。このステップS17において、ブロックc3がオブジェクト内であると判断した場合はステップS18に進み、そうでないと判断した場合はステップS19に進む。

【0281】ステップS18では、バッファより1ビットを取り出し、ブロックc3の符号化パターンの判断に使用する。

【0282】ステップS19では、図37に示したブロックc4がオブジェクト内であるか否かを判断する。このステップS19において、ブロックc4がオブジェクト内であると判断した場合はステップS20に進み、そうでないと判断した場合はステップS21に進む。

【0283】ステップS20では、バッファより1ビットを取り出し、ブロックc4の符号化パターンの判断に使用する。

【0284】ステップS21では、図35の処理に戻る。

【0285】次に、図38には、フラグ(cbpcb)のビット割り当ての例を示す。

【0286】図38の(a)は、4:2:2フォーマットの場合のオブジェクト内のブロックの数及びその発生ビット数を示し、図38の(b)は、4:4:4フォーマットでの発生ビット数及びその割り当てビット順の例を示す。

【0287】同様に4:2:2フォーマット、4:4:4フォーマットにおいて、フラグ(cbpc)が「01」、「11」の場合、色差CrのブロックにDCT係数を持つブロックが存在するものと判断され、4:2:2フォーマットの場合は2ビット、4:4:4フォーマットの場合は4ビットの色差Crのブロックの符号化状態を表すフラグ(cbpcr)が読み込まれる。

【0288】これらのフラグは、フラグ(cbpcb)の場合と同様の方法により符号化/復号化される。これらの符号化法は前述したフラグ(cbpcb)の場合と同様にそのオブジェクト内のブロック数に応じて変化し、それぞれのビットは該当ブロックのDCT係数の状態を示す。

【0289】このようにして、各色差ブロックに対してDCT係数の有無が判別される。この結果は、後のDCT係数の出力/入力過程において使用される。

【0290】また、マクロブロック内のブロックカウント(block_count)は、マクロブロック内でのオブジェクト内に位置するブロックの個数、すなわち(オブジェクト内の輝度のブロックの個数+オブジェクト内の色差Cbのブロックの個数+オブジェクト内の色差Crのブロックの個数)となる。

【0291】ブロックカウント(block_count)の最大値は、色差フォーマットにより決定されるものとし、図39に示すように定められる。

【0292】次に、図示は省略するが、マクロブロック内の各ブロックの符号化方法を表すシンタックスと、DCT係数の有無の判別結果の使用方法について説明する。

【0293】ここで、便宜上これらの判別結果は、当該シンタックスの配列パターンコード(pattern_code)に格納されるものとする。例えばパターンコード[i](pattern_code[i])は、i番目のブロックの符号化状態を示すものとする。パターンコード[1](pattern_code[1])

は、前記図3の(a)～(c)に示したブロック1のDCT係数の状態を示し、パターンコード[5] (pattern_code[5]) は、同様にブロック5の状態を示す。このパターンコードの配列の長さは色差フォーマットにより異なり4:2:0フォーマットの場合は0から5、4:2:2フォーマットで0から7、4:4:4フォーマットで0から11の要素を持つものとする。それぞれの要素は、対応するブロックがDCT係数を持つ場合には1、それ以外に場合には0をとる。また、DCTの読み出しを行うか否かは、当該シンタックスの先頭の行において求められた値により決定される。また、iがオブジェクトの外側のブロックであった場合、その番号は欠番とされ、パターンコードに格納されるビットの総数は、オブジェクト内のブロックの個数と一致する。

【0294】次に、B-VOPでの色差信号の符号化パターンの認識方法を説明する。

【0295】図30及び図31に示したマクロブロックのシンタックスに戻り、B-VOPにおいての符号化ブロックのパターンの認識方法を説明する。

【0296】B-VOPでは、そのマクロブロックの符号化パターンを示すのに、当該マクロブロックのシンタックスにおけるフラグ (cbpb) が使われる。当該フラグ (cbpb) は、4:2:0フォーマットの時には、MPEG4のビジュアルコミッティドラフト (Visual Comitee Draft) と同様の定義である。

【0297】以下、クロマタイプ (chroma_type) が「10」 (4:2:0フォーマット) の時のフラグ (cbpb) の定義、すなわちMPEG4のビジュアルコミッティドラフトにおけるフラグ (cbpb) を説明する。

【0298】フラグ (cbpb) は、前述した通り6ビットのフラグであり、その値はマクロブロック内の各ブロックの符号化状態を示す。この時、フラグ (cbpb) の各ビットは、先頭のビットより順に図3の(a)に示したブロック0, 1, 2, 3, 4, 5がDCT係数を持っているか否かを示し、そのビットが1のとき対応するブロックにおいてDCT係数が存在すると判断され、0のとき存在しないと判断される。ここで読み込まれたフラグ (cbpb) は、後段においてDCT係数の入出力に使用される。また、I-VOP、P-VOPの場合と同様、このビット数は、あるマクロブロックに存在するオブジェクト内のブロックの個数と一致するものとする。すなわちオブジェクト内のブロックが3個であった場合、このビットは3ビットとなり、順にオブジェクト内のブロックの符号化状態を示すフラグとして利用される。

【0299】これらの判断結果の使用方法は、B-VOPの場合においても、I-VOP、P-VOPと同様である。

【0300】また、フラグ (cbpb) の下位2ビットは、前述したフラグ (cbpc) と全く同様に色差ブロックの符号化状態を示す。そのため、4:2:2フォーマットお

より4:4:4フォーマットにおける色差ブロックの符号化パターンの判別は、この下位2ビットおよびクロマタイプ (chroma_type) を用い、I-VOP、P-VOPの時と同様の手段を用いて行われる。これらの下位2ビットが1を示した場合に読み込まれるビット数も符号化を行っている画像の色差フォーマット、及びオブジェクト内のブロックの個数により変化する。

【0301】このようにして、色差フォーマットを用いること、および色差の符号化パターンを判断する手段を

10 用いることにより、4:2:0フォーマット、4:2:2フォーマット、4:4:4フォーマットの各色差フォーマットの符号化/復号化が可能となる。次に、MBバイナリシェイプコーディング (mb_binary_shape_coding) のシンタックス、モーションベクター (motion_vector)、ブロック (block) の各シンタックスについても図示は省略するが、ブロック (block) においては、DCT係数の差分値、及びその値が符号化される。また、サブシーケンスDCTコイフシャンツ (SubsequentDCTcoefficients) は、その他のAC係数を示し、これもVL C符号化される。

【0302】モーションシェイプテクスチャ (motion_shape_texture) のシンタックスに戻って、データパーティショニング (data_partitioning) のフラグが1であるかつテクスチャ情報が伝送される場合には、図示しないデータパーティショニングモーションシェイプテクスチャ (data_partitioning_motion_shape_texture) が符号化される。このデータパーティショニングモーションシェイプテクスチャ (data_partitioning_motion_shape_texture) は大きく2つの部分から構成されており、ブロック (block) のシンタックスにおけるデータパーティショニングI-VOP (data_partitioning_I_VOP) 及びデータパーティショニングP-VOP (data_partitioning_P_VOP) から構成される。

【0303】これらデータパーティショニングI-VOP (data_partitioning_I_VOP) 及びデータパーティショニングP-VOP (data_partitioning_P_VOP) とも、VOPの性質を示すフラグ群及び各ブロックのDCT係数を符号化する。DCT係数の符号化方法は、前述したDCT係数の符号化方法と同様の方法で行う。

【0304】以上説明した本発明実施の形態のエンコーダおよびデコーダは、それ専用のハードウェアによって実現することもできるし、例えばコンピュータに上述したような処理を行わせるためのプログラムを実行させることによっても実現することができる。

【0305】次に、本発明の第2の実施の形態について説明する。

【0306】この第2の実施の形態では、前述した実施の形態 (第1の実施の形態) においてB-VOPにおける色差ブロックの符号化パターンを、フラグ (cbpb) のビット長をクロマタイプ (chroma_type) により変化さ

せて求めるように変更したものである。すなわち、第2の実施の形態においては、マクロブロックの符号化を行うそのシンタックス及び、符号化／復号化方法のみが第1の実施の形態と異なる。

【0307】それ以外の符号化／復号化方法、シンタックス等は第1の実施の形態と同様である。

【0308】第1の実施の形態においてフラグ (cbpb) は常に6ビットのフラグであり、その下位2ビットが常に色差ブロックの状態を示していたが、これを第2の実施の形態では4:2:2フォーマットの場合には最大8ビットのフラグ、4:4:4フォーマットの場合には最大12ビットのフラグとする。

【0309】この第2の実施の形態におけるマクロブロックの符号化方法を示すシンタックスの一部（前半部分）を図40及び図41に示す。

【0310】4:2:2フォーマットの場合、この8ビットのフラグは、先頭から図3の(b)に示したブロック0, 1, 2, 3, 4, 5, 6, 7のDCT係数の有無を表すものとする。

【0311】同様に4:4:4フォーマットの場合、1 20 2ビットのフラグの先頭のビットより図3の(c)に示したブロック0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11に対応しているものとし、そのDCT係数の有無を表すものとする。

【0312】4:2:2フォーマット、4:4:4フォーマットとも、対応するブロックがオブジェクトの外側に位置する時、そのブロックに対応するフラグは存在せず、その場合、フラグ (cbpb) のビット長はそのマクロブロックにおいて、オブジェクト内に存在するブロックの個数と等しくなる。

【0313】このようにして、色差フォーマットを用いること、および色差の符号化パターンを判断する手段を用いることにより、4:2:0フォーマット、4:2:2フォーマット、4:4:4フォーマットの色差フォーマットの符号化／復号化が可能となる、この第2の実施の形態においてもエンコーダおよびデコーダは、それ専用のハードウェアによって実現することもできるし、コンピュータに、上述したような処理を行わせるためのプログラムを実行させることによっても実現することができる。

【0314】なお、本発明の提供媒体は、上述した符号化データを提供する場合のみならず、本発明実施の形態のエンコーダやデコーダの動作を行わせるためのプログラムを提供するものとすることも可能である。

【0315】以上説明したように、本発明の各実施の形態においては、画像の色差フォーマットを示すフラグを用いること、また色差のブロックの符号化パターンを上記した手段を用いて示すことにより、4:2:0フォーマット以外の色差フォーマットの符号化／復号化が可能となる。

【0316】

【発明の効果】本発明によれば、画像の輝度成分と色差成分とをそれぞれブロック化して各ブロック毎の動きベクトルを生成する際に、上記輝度成分のブロックに対して得られた動きベクトルに基づいて、対応する色差成分のブロックの動きベクトルを生成することにより、4:2:0フォーマット以外の色差フォーマットにも好適な動きベクトル生成方法、画像符号化装置、動き補償方法、及び動き補償装置を提供することができる。

【0317】また、上記色差成分として、複数種類の色差フォーマットのいずれかが選択された色差成分を用い、この色差フォーマットの種類に応じて、上記輝度成分のブロックに対して得られた動きベクトルに基づく色差成分のブロックの動きベクトルの計算を行うことにより、4:2:2フォーマット、4:4:4フォーマットのような色差成分の解像度が高いフォーマットの画像信号についての動きベクトル検出や動き補償が有効に行え、色差成分の高い解像度を活かした符号化／復号化が可能となる。

【図面の簡単な説明】

【図1】本発明実施の形態のエンコーダの概略構成を示すブロック回路図である。

【図2】ビデオオブジェクトシークエンスからブロックまでの構成説明に用いる図である。

【図3】色差フォーマットにおけるマクロブロック内のブロック配置の説明に用いる図である。

【図4】背景画像とオブジェクト画像との関係説明に用いる図である。

【図5】図1のエンコーダのV.O.P符号化部の詳細な構成例を示すブロック回路図である。

【図6】輝度の形状情報から色差の形状情報を作成する作成方法の説明に用いる図である。

【図7】図5のパディング処理器の詳細な構成例を示すブロック回路図である。

【図8】水平方向のパディング処理の手順を表す図である。

【図9】垂直方向のパディング処理の手順を表す図である。

【図10】拡張パディング処理されるマクロブロックの説明に用いる図である。

【図11】拡張パディング処理の説明に用いる図である。

【図12】パディングされるマクロブロックが複数のオブジェクトの境界を持つマクロブロックに囲まれている場合に、参照するマクロブロックの順番を示す図である。

【図13】マクロブロックに対する動き補償動作を説明するためのフローチャートである。

【図14】4:2:0フォーマットの場合の色差ブロックの動きベクトルを説明するための図である。

【図15】4:2:2フォーマットの場合の色差ブロックの動きベクトルを説明するための図である。

【図16】4:4:4フォーマットの場合の色差ブロックの動きベクトルを説明するための図である。

【図17】色差ブロックの解像度変換のためのテーブルを示す図である。

【図18】オーバーラップ動き補償の動作を説明するための図である。

【図19】オーバーラップ動き補償に用いられる重みテーブルを示す図である。

【図20】オーバーラップ動き補償に用いられる重みテーブルを示す図である。

【図21】図5のDCT係数差分化器の詳細な構成例を示すブロック回路図である。

【図22】図21のDCT成分予測値選択器の動作説明に用いる図である。

【図23】予測に用いられるDCT係数(AC係数及び各DC係数)の説明に用いる図である。

【図24】各色差フォーマットにおいてAC/DC予測に用いるブロックの説明を行うための図である。

【図25】色差ブロックにおいてオブジェクト内のブロックかどうか判別する方法についての説明に用いる図である。

【図26】図1のエンコーダから出力されるビットストリームを復号する本実施の形態のデコーダの構成例を示すブロック回路図である。

【図27】図26のVOP復号部の詳細な構成例を示すブロック回路図である。

【図28】VOL(video Object Layer)のシンタックスにおいて、ビデオオブジェクトレイヤスタートコード(video_object_layer_start_code)に続く部分から構成されるシンタックスの前半部分を示す図である。

【図29】クロマフォーマット(chroma_format)の定義説明に用いる図である。

【図30】マクロブロックのシンタックスの一部を示す図である。

【図31】図30のシンタックスの続きを示す図である。

【図32】I-VOPのフラグ(mbpc)のVLCテーブルを示す図である。

【図33】P-VOPのフラグ(mbpc)のVLCテーブルを示す図である。

【図34】マクロブロックタイプ(MBtype)の定義説明に用いる図である。

【図35】フラグ(cbpcb)の定義の手順のうち前半部の手順を示すフローチャートである。

【図36】フラグ(cbpcb)の定義の手順のうち後半部の手順を示すフローチャートである。

* 【図37】図35及び図36中で使用するブロックの説明に用いる図である。

【図38】フラグ(cbpcb)のビット割り当ての例の説明に用いる図である。

【図39】ブロックカウント(block_count)の最大値の説明に用いる図である。

【図40】第2の実施の形態におけるマクロブロックの符号化方法を示すシンタックスの一部を示す図である。

10 【図41】図40のシンタックスの続きを示す図である。

【図42】一般的なMPEGエンコーダの概略構成を示すブロック回路図である。

【図43】一般的なMPEGデコーダの概略構成を示すブロック回路図である。

【図44】4:2:0フォーマットの説明に用いる図である。

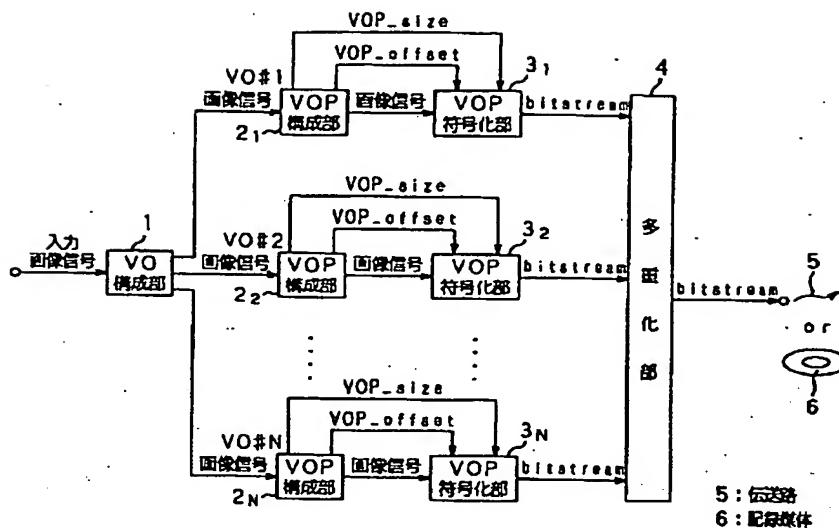
【図45】4:2:2フォーマットの説明に用いる図である。

20 【図46】4:4:4フォーマットの説明に用いる図である。

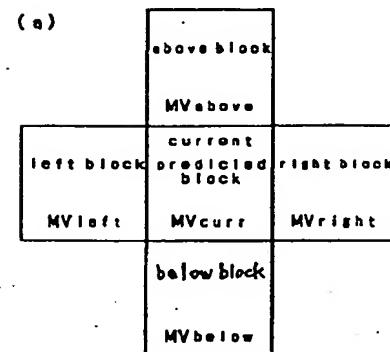
【符号の説明】

1 VOP構成部、 2 VOP構成部、 3 VOP符号化部、 4 多重化部、 5 伝送路、 6 記録媒体、 31, 41 フレームメモリ、 32動きベクトル検出器、 33, 40 演算器、 34 DCT器、 35 量子化器、 36 VLC器、 37 バッファ、 38 逆量子化器、 39 IDCT器、 42 動き補償器、 44 DCT係数差分化器、 50 形状情報符号化器、 51 パディング処理器、 52 形状情報復号器、 53 色差用形状情報作成器、 61 水平方向パディング処理器、 62 垂直方向パディング処理器、 63 拡張パディング器、 71 逆多重化器、 72 VOP復号部、 73 画像再構成部、 74 モニタ、 101 バッファ、 102 VLC器、 103 逆多重化器、 104 IDCT器、 105 演算器、 106 フレームメモリ、 107 動き補償器、 108 パディング処理器、 109 色差用形状情報作成器、 110 形状情報復号器、 111 DCT係数逆差分化器、 400 イントラフラグ発生器、 401スイッチ、 402 DCT係数予測器、 403 DCT係数予測値選択器、 404 DCT係数予測モード蓄積メモリ、 405 逆量子化器、 方位表示装置、 2 CPU、 3 RAM、 4 ROM、 5 バス、 6 A/Dコンバータ、 7 出力ポート、 8 地磁気センサ、 9 LED表示器、 10 地磁気センサ入力プログラム、 11 方位決定プログラム、 12 表示プログラム、 13 演算制御部、 14 データテーブル

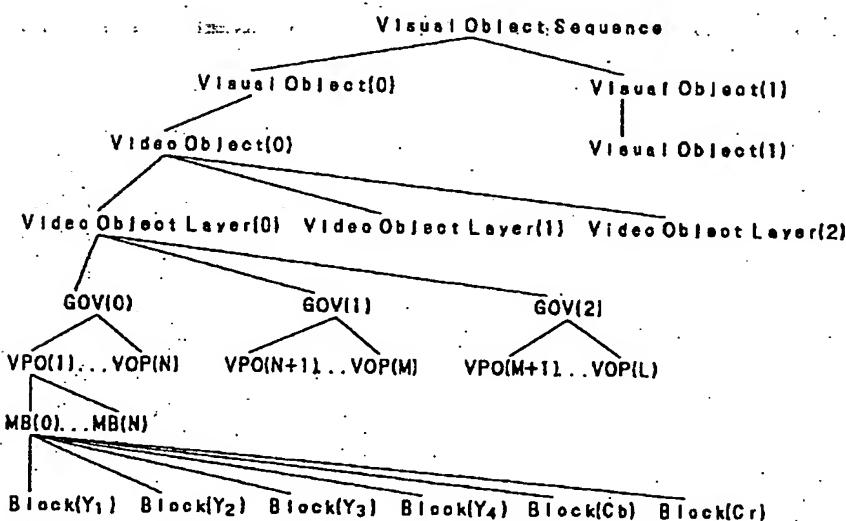
【図1】



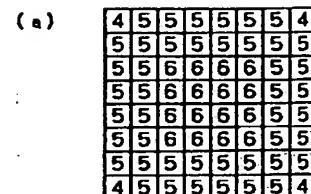
【図18】



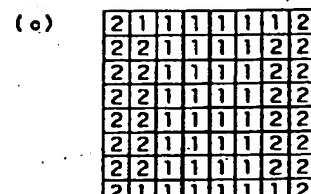
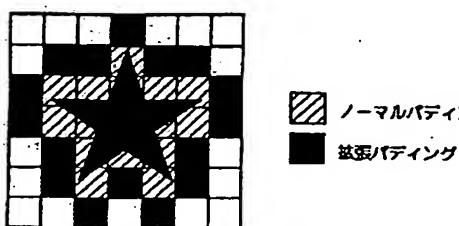
【図2】



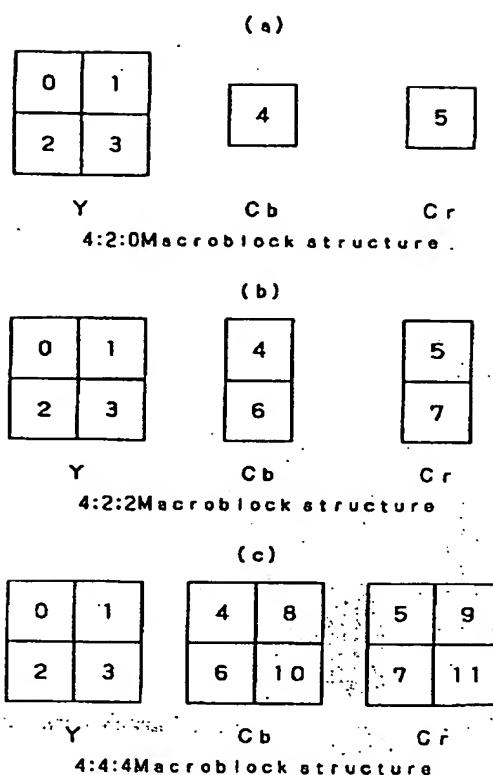
【図19】



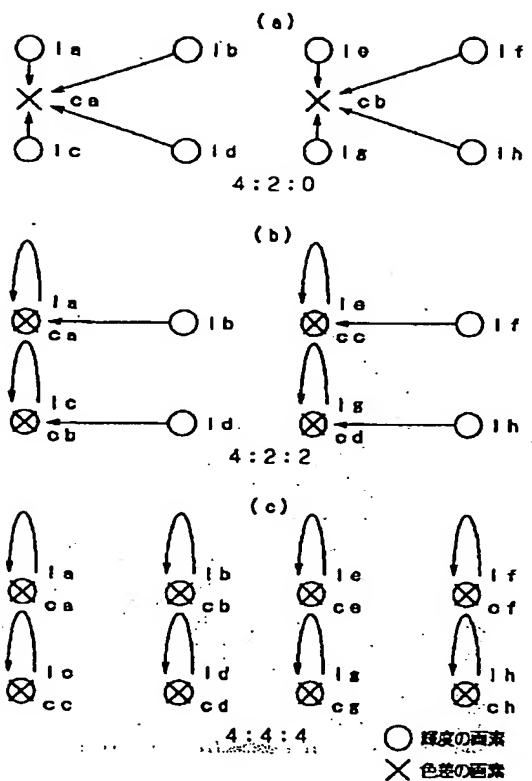
【図10】



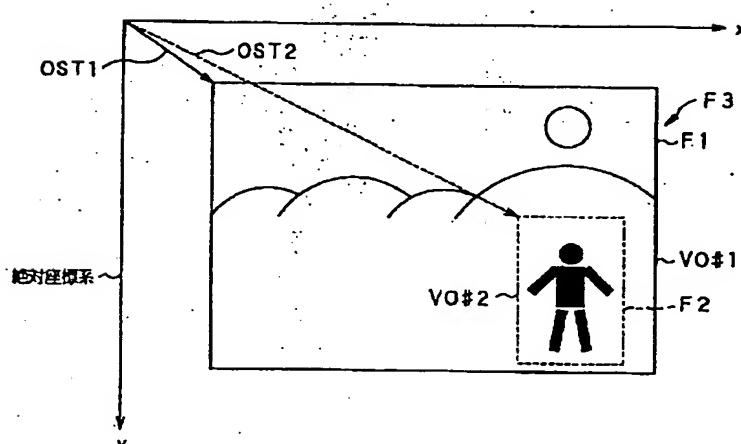
【図3】



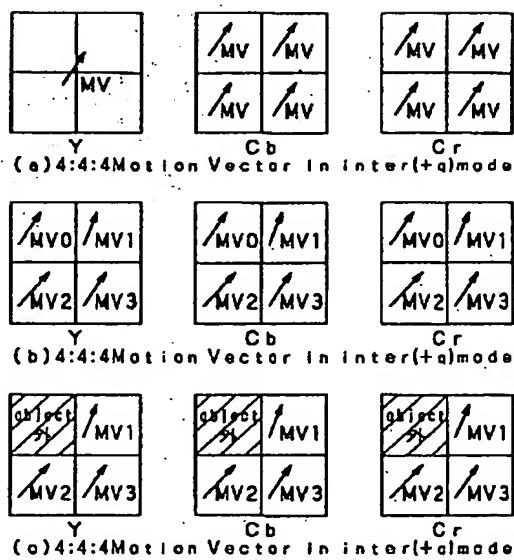
【図6】



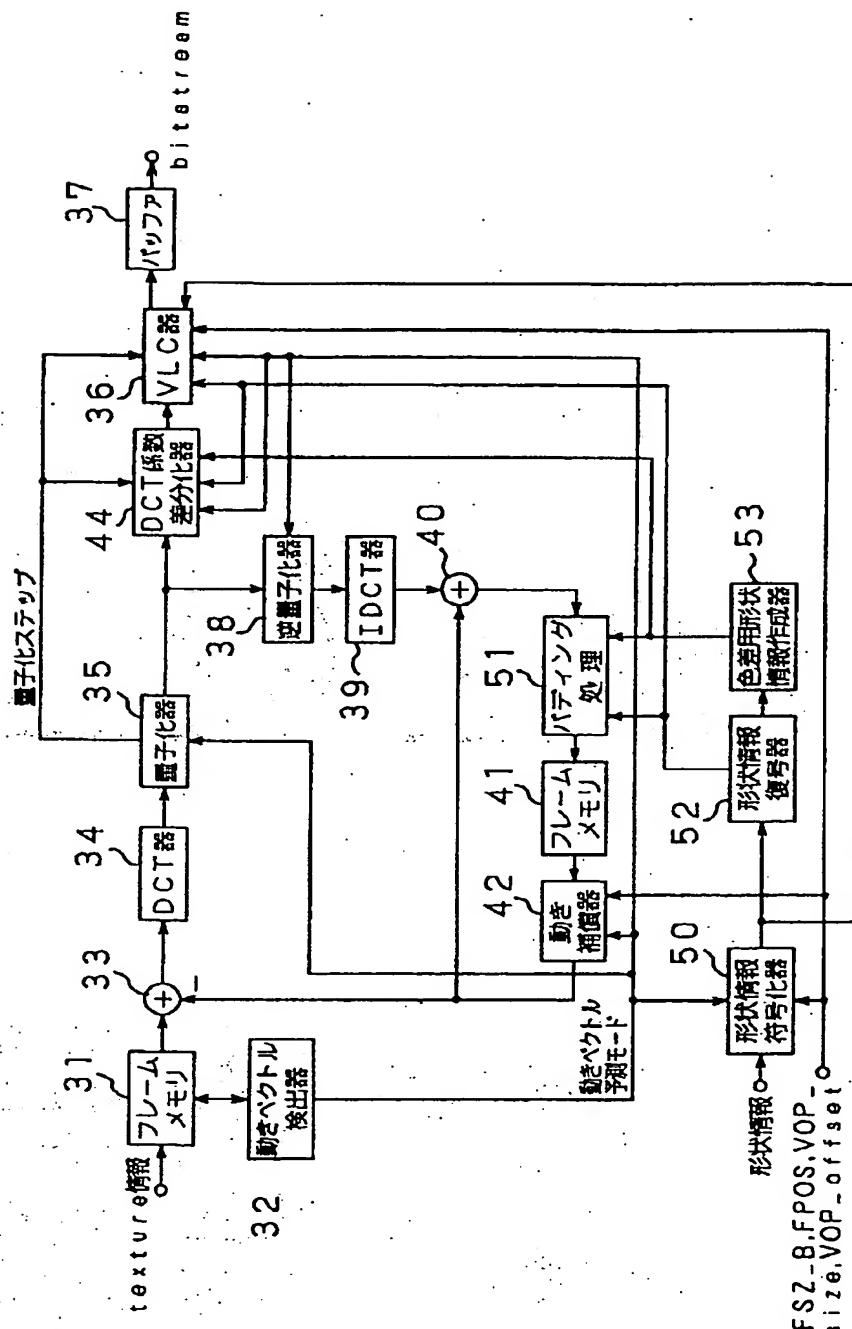
【図4】



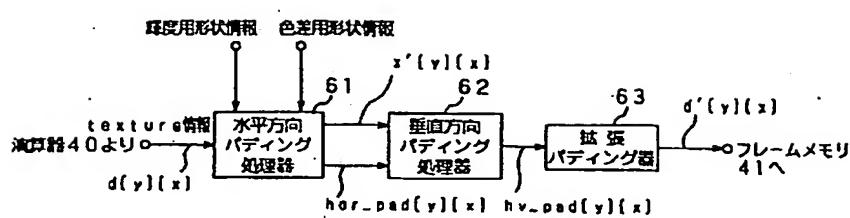
【図16】



〔図5〕



【図7】



【図29】

Meaning of chroma_format

chroma_format	Meaning
00	reserved
01	4:2:0
10	4:2:2
11	4:4:4

【図8】

```

for (x=0;x<N;x++) {
    if(s[y][x]==1) {hor_pad[y][x]=d[y][x];s'[y][x]=1;}
    else {
        if(s[y][x]==1&&s[y][x']==1)
            hor_pad[y][x]=(d[y][x']+d[y][x'])//2;
        s'[y][x]=1;
    } else if(s[y][x]==1) {
        hor_pad[y][x]=d[y][x];s'[y][x]=1;
    } else if(s[y][x']==1) {
        hor_pad[y][x]=d[y][x'];s'[y][x]=1;
    }
}
}

```

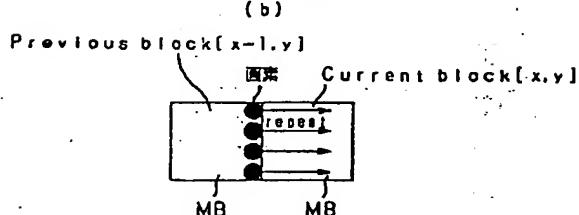
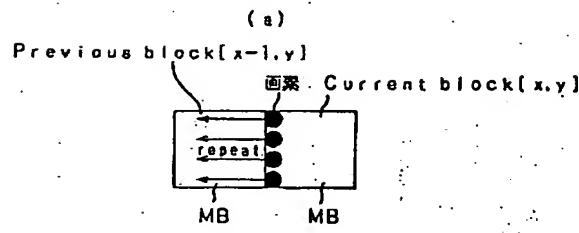
```

for (y=0;y<M;y++) {
    if(s'[y][x]==1)
        hv_pad[y][x]=hor_pad[y][x];
    else {
        if(s'[y][x]==1&&s'[y][x]==1)
            hv_pad[y][x] = (hor_pad[y][x] +
hor_pad[y'][x])//2;
        else if(s'[y][x]==1)
            hv_pad[y][x]=hor_pad[y][x];
        else if(s'[y][x]==1)
            hv_pad[y][x]=hor_pad[y'][x];
    }
}
}

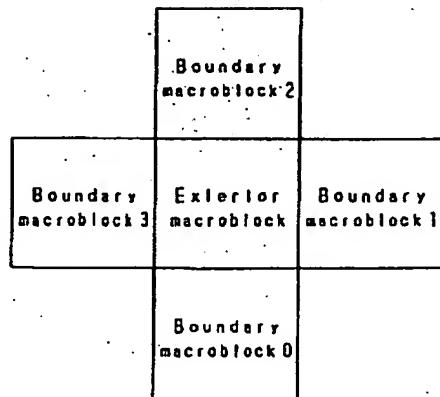
```

【図9】

【図11】



【図12】

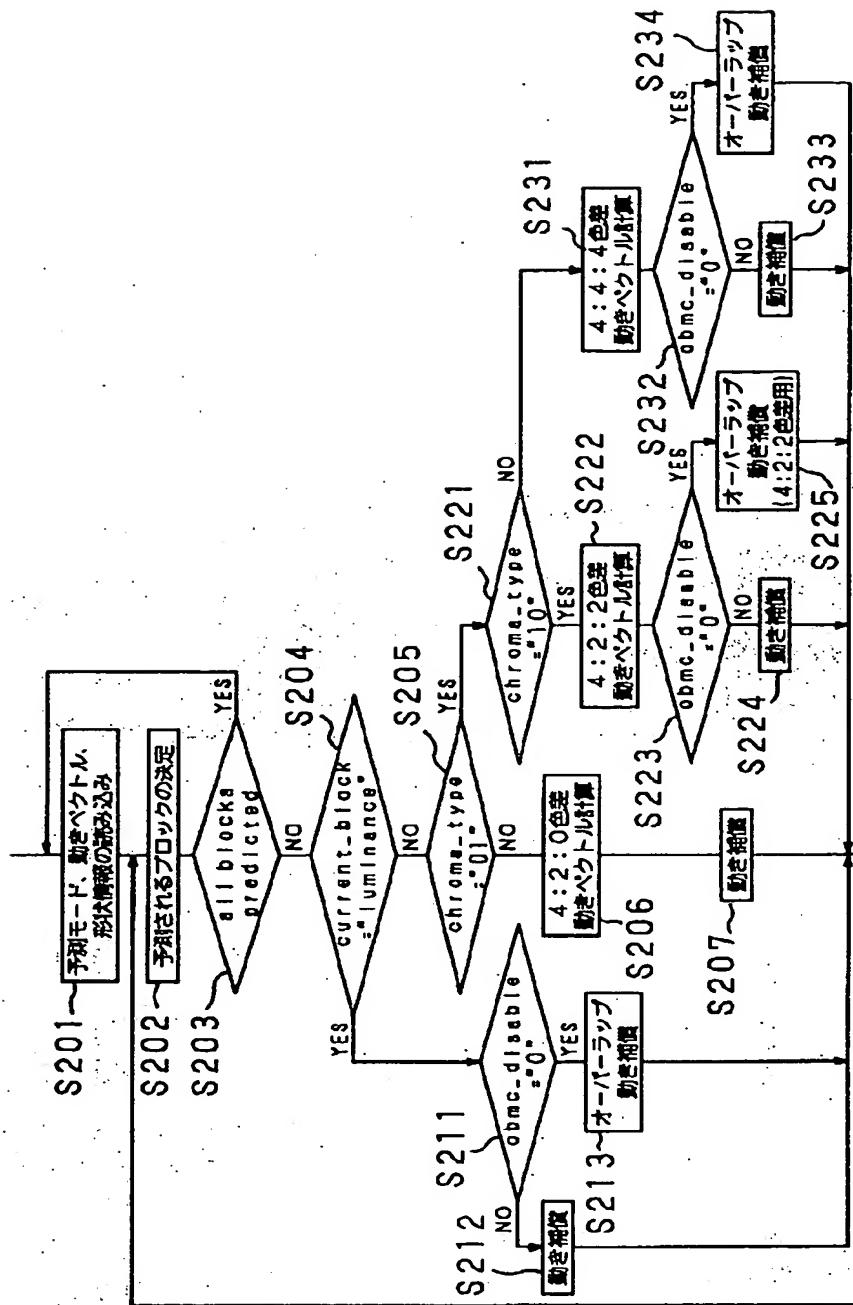


【図39】

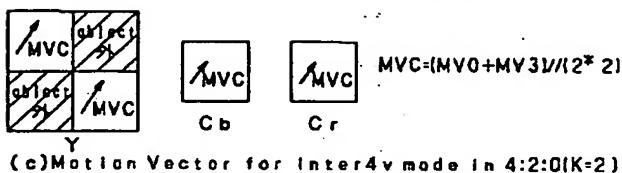
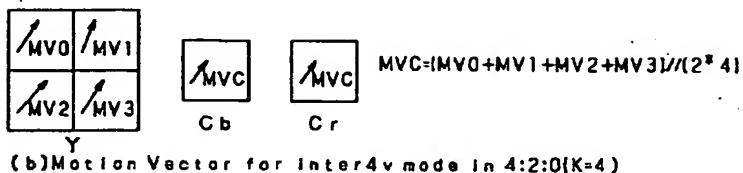
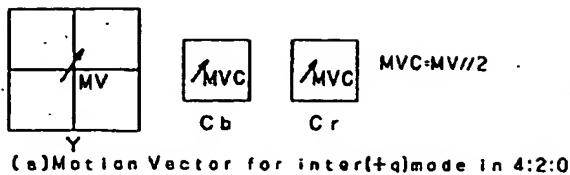
block_count as a function of chroma_format

chroma_format	block_count
4:2:0	6
4:2:2	8
4:4:4	12

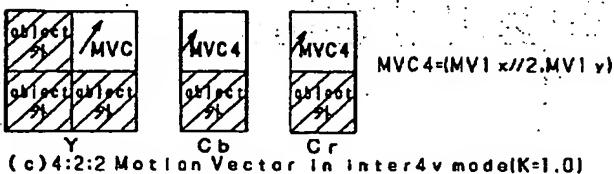
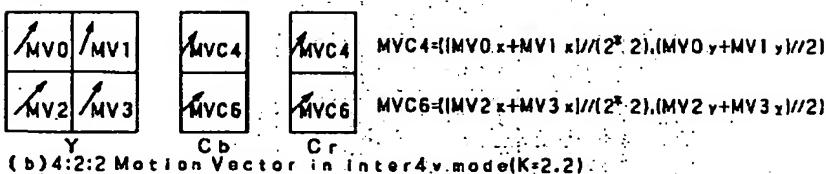
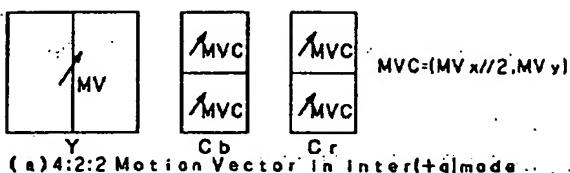
【図13】



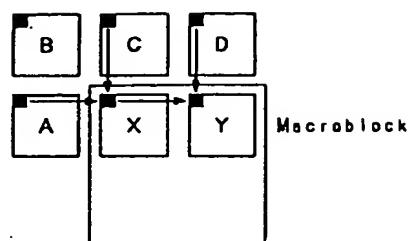
【図14】



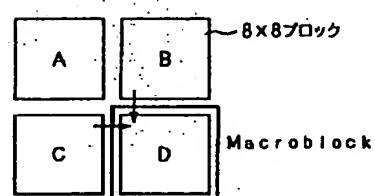
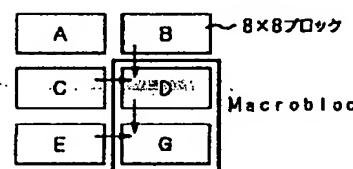
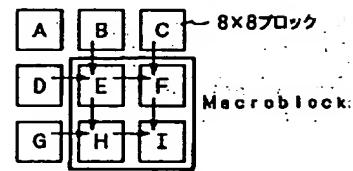
【図15】



【図22】



【図24】



【図32】

Code	mbtype	cdpc (Cb,Cr)
1	3	00
001	3	01
010	3	10
011	3	11
0001	4	00
0000 01	4	01
0000 10	4	10
0000 11	4	11
0000 0000 1	Stuffing	-

【図17】

(a)

sixteenth pixel position	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	//16
resulting position	0	0	0	1	1	1	1	1	1	1	1	1	1	1	2	2	//2

(b)

twelfth pixel position	0	1	2	3	4	5	6	7	8	9	10	11	//12
resulting position	0	0	0	1	1	1	1	1	1	1	2	2	//2

(c)

eight pixel position	0	1	2	3	4	5	6	7	//8
resulting position	0	0	1	1	1	1	1	2	//2

(d)

fourth pixel position	0	1	2	3	//4
resulting position	0	1	1	1	//2

【図33】

Code	mbtype	cdpc (Cb,Cr)
1	0	00
0011	0	01
0010	0	10
0001 01	0	11
011	1	00
0000 111	1	01
0000 110	1	10
0000 0010 1	1	11
010	2	00
0000 101	2	01
0000 100	2	10
0000 0101	2	11
0001 1	3	00
0000 0100	3	01
0000 0011	3	10
0000 0011	3	11
0001 00	4	00
0000 0010 0	4	01
0000 0001 1	4	10
0000 0001 0	4	11
0000 0000 1	Stuffing	-

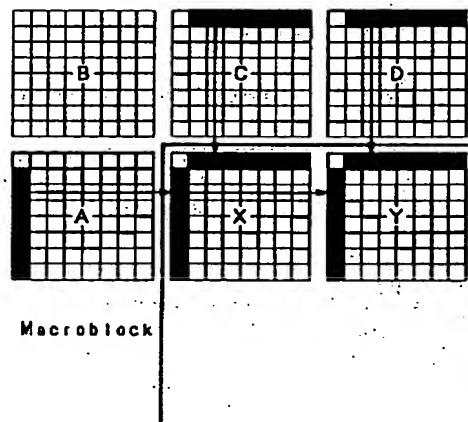
【図20】

(a)

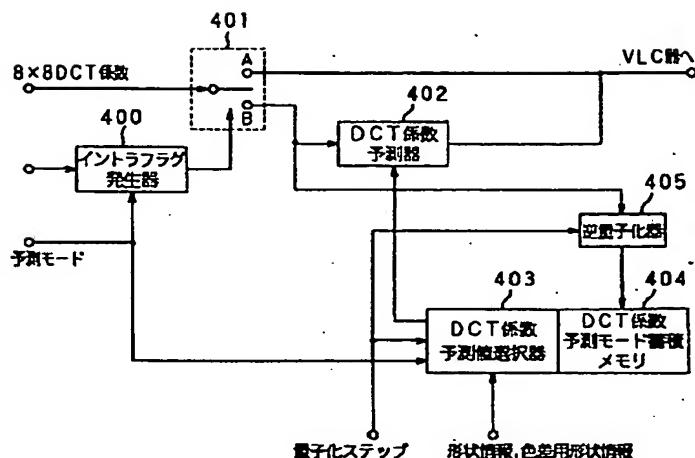
5	5	5	5	5	5	5	5
5	5	5	5	5	5	5	5
6	6	6	6	6	6	6	6
6	6	6	6	6	6	6	6
6	6	6	6	6	6	6	6
6	6	6	6	6	6	6	6
6	6	6	6	6	6	6	6
5	5	5	5	5	5	5	5
5	5	5	5	5	5	5	5

(b)

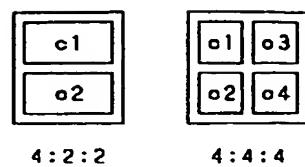
3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3
2	2	2	2	2	2	2	2
2	2	2	2	2	2	2	2
2	2	2	2	2	2	2	2
2	2	2	2	2	2	2	2
3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3



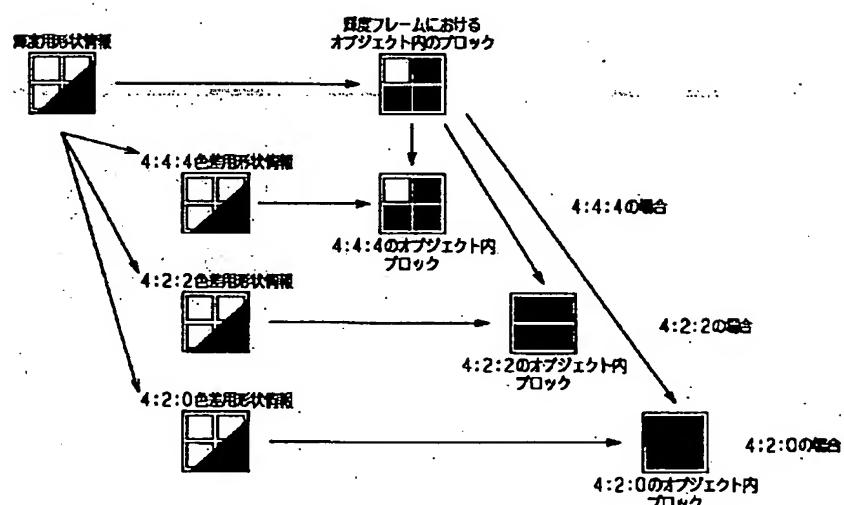
【図21】



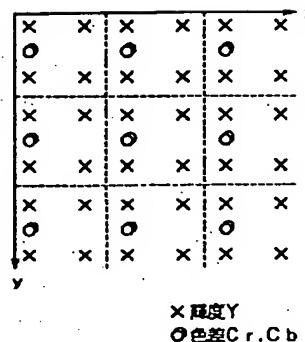
【図37】



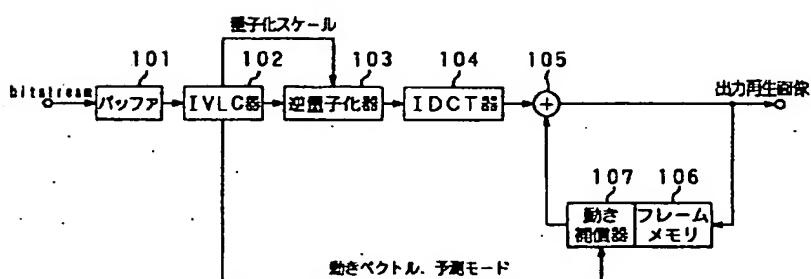
【図25】



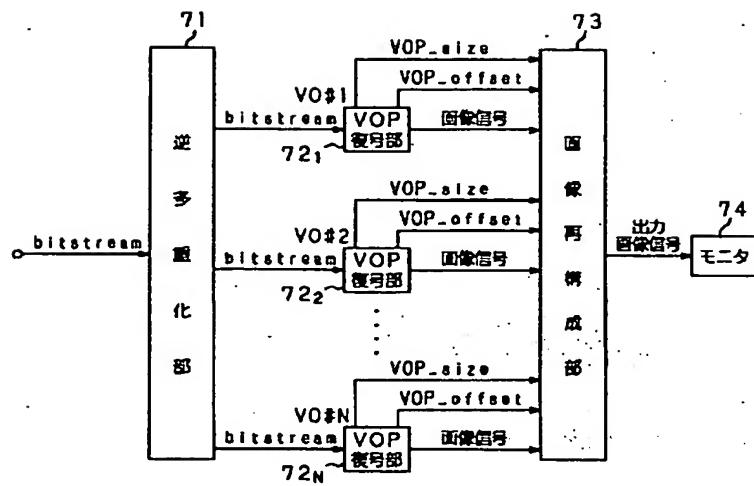
【図44】



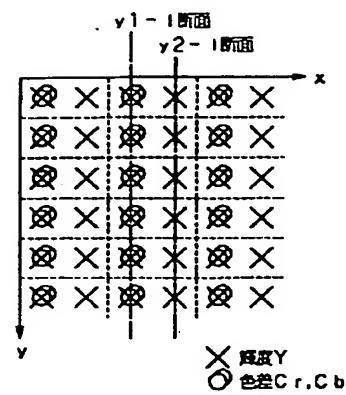
【図43】



【図26】



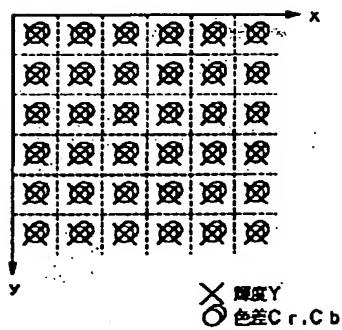
【図45】



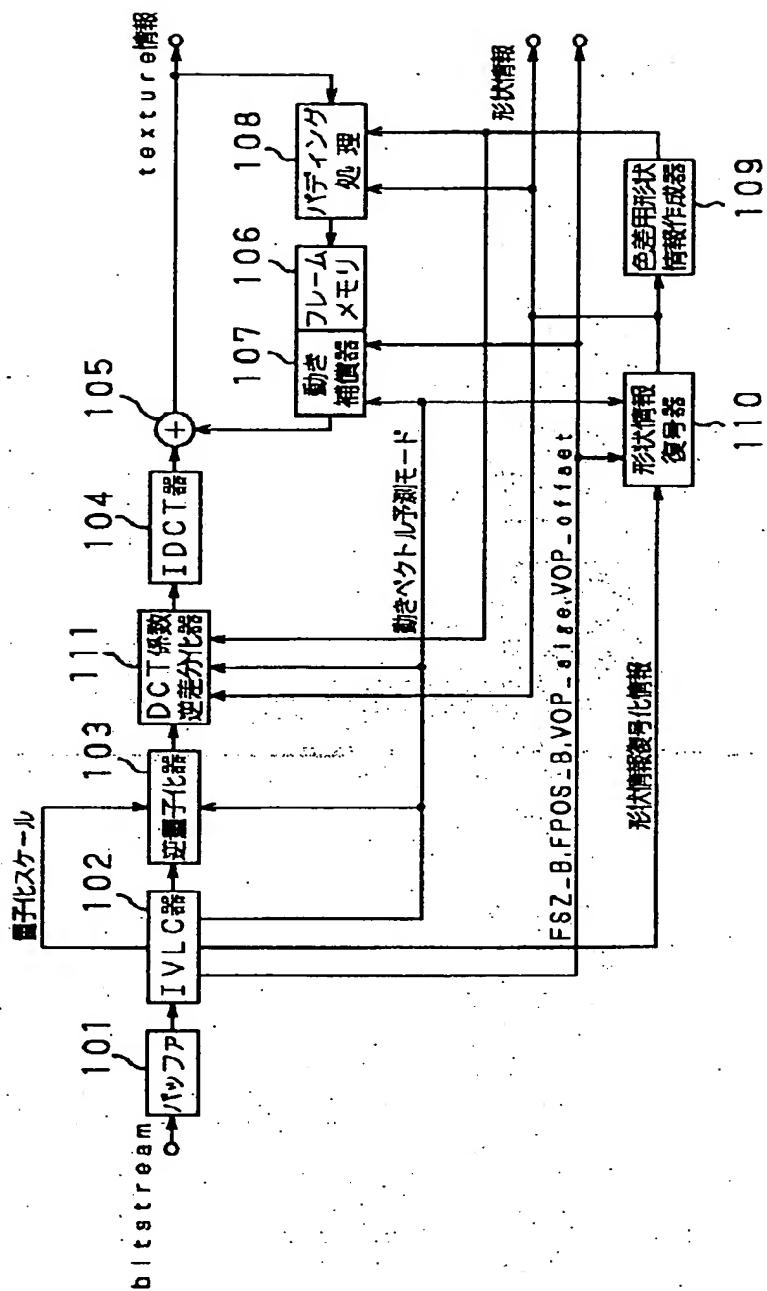
【図34】

VOP type	mb type	Name	not_coded	mcfc	cbpy	dquant	mvd	mvd3-4
P	not coded	-	1				1	
P	0	inter	1	1	1			
P	1	Inter+q	1	1	1		1	
P	2	Inter4v	1	1	1		1	1
P	3	Intra	1					
P	4	Intra+q	1				1	
P	stuffing	-	1					
I	3	Intra			1			
I	4	Intra+q			1		1	
I	stuffing	-	1					
S(update)	not coded	-	1					
S(update)	0	Inter	1	1	1			
S(update)	1	Inter+q	1	1	1		1	
S(update)	3	Intra	1	1	1			
S(update)	4	Intra+q	1	1	1		1	
S(update)	stuffing	-	1	1				
S(piece)	3	Intra		1	1			
S(piece)	4	Intra+q		1	1		1	
S(piece)	stuffing	-	1					

【図46】



【図27】



[図28]

Video Object Layer

VideoObjectLayer()	No.of bits	Mnemonic
if(nextbits()==video_object_layer_start_code) {		
short_video_header=0		
video_object_layer_start_code	32	bslbf
is_object_layer_identifier	1	uimsbf
if(is_object_layer_identifier) {		
video_object_layer_verid	4	uimsbf
video_object_layer_priority	3	uimsbf
}		
vol_control_parameters	1	bslbf
if(vol_control_parameters)		
aspect_ratio_info	4	uimsbf
VOP_rate_code	4	uimsbf
bit_rate	30	uimsbf
vbr_buffer_size	18	uimsbf
low_delay	1	uimsbf
}		
chroma_format	2	
video_object_layer_shape	2	uimsbf
VOP_time_increment_resolution	15	uimsbf
fixed_VOP_rate	1	bslbf
if(video_object_layer_shape=="binary only") {		
if(video_object_layer_shape=="rectangular") {		
marker_bit	1	bslbf
video_object_layer_width	13	uimsbf
marker_bit	1	bslbf
video_object_layer_height	13	uimsbf
}		
obmc_disable	1	bslbf
sprite_enable	1	bslbf
if(sprite_enable) {		
sprite_width	13	uimsbf
marker_bit	1	bslbf
sprite_height	13	uimsbf
marker_bit	1	bslbf
sprite_left_coordinate	13	simsbf
marker_bit	1	bslbf
sprite_top_coordinate	13	simsbf
marker_bit	1	bslbf
no_of_sprite_warping_points	6	uimsbf
sprite_warping_accuracy	2	uimsbf

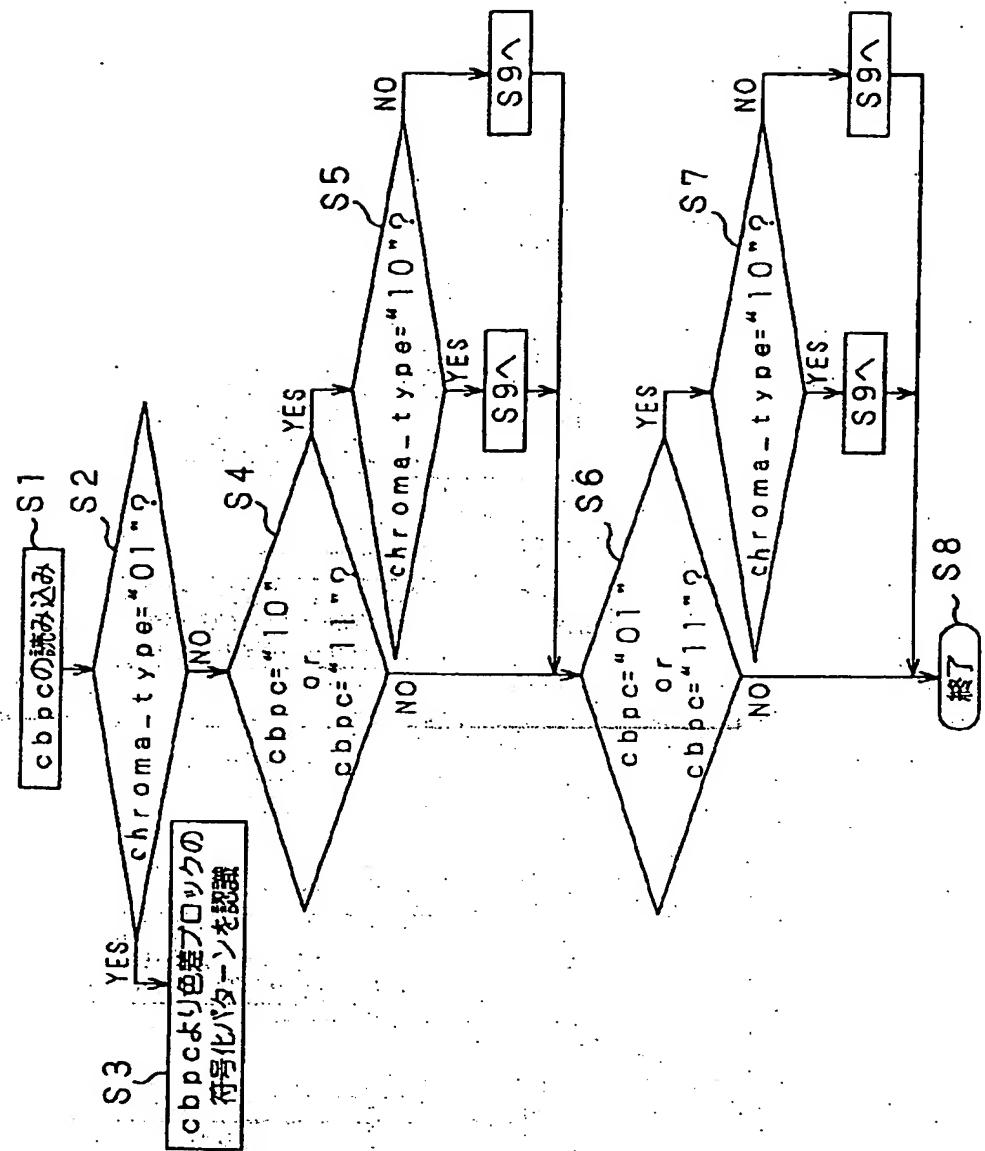
【図30】

macroblock()	No.of bits	Mnemonic
if(VOP_coding_type != "B") {		
if(video_object_layer_shape != "rectangular" &&!(sprite_enable && low_latency_sprite_enable && sprite_transmit_mode == "update"))		
mb_binary_shape_coding()		
if(video_object_layer_shape == "binary only") {		
if(transparent_mb()) {		
if(VOP_coding_type == "T" &&!(sprite_enable && sprite_transmit_mode == "piece"))		
not_codeb	1	bslbf
if(not_coded & VOP_coding_type == "I") {		
mcbpc	1-9	vicbf
if(cbpc == "10" cbpc == "11") {		*
if(chroma_format == "10") {		*
cbpcb	1-2	vicbf
if(chroma_format == "11") {		*
cdpcb	1-4	vicbf
if(cdpc == "01" cdpc == "11") {		*
if(chroma_format == "10") {		*
cbper	1-2	vicbf
if(chroma_format == "11") {		*
cbper	1-4	vicbf
if((!short_video_header && derived_mb_type == 31 derived_mb_type != 4)) {		*
ac_pred_flag	1	bslbf
if(derived_mb_type == "stuffing") {		
cbpy	1-6	vicbf
else {		
return();		
if(derived_mb_type == 11 derived_mb_type != 4) {		
dquant	2	uimsbf
if(interlaced) {		
interlaced_information()		
if(!(ref_select_code == "11" && scalability) && VOP_coding_type != "S") {		
if(derived_mb_type == 01 derived_mb_type == 1) {		
motion_vector("forward")		
if(field_prediction) {		
motion_vector("forward")		

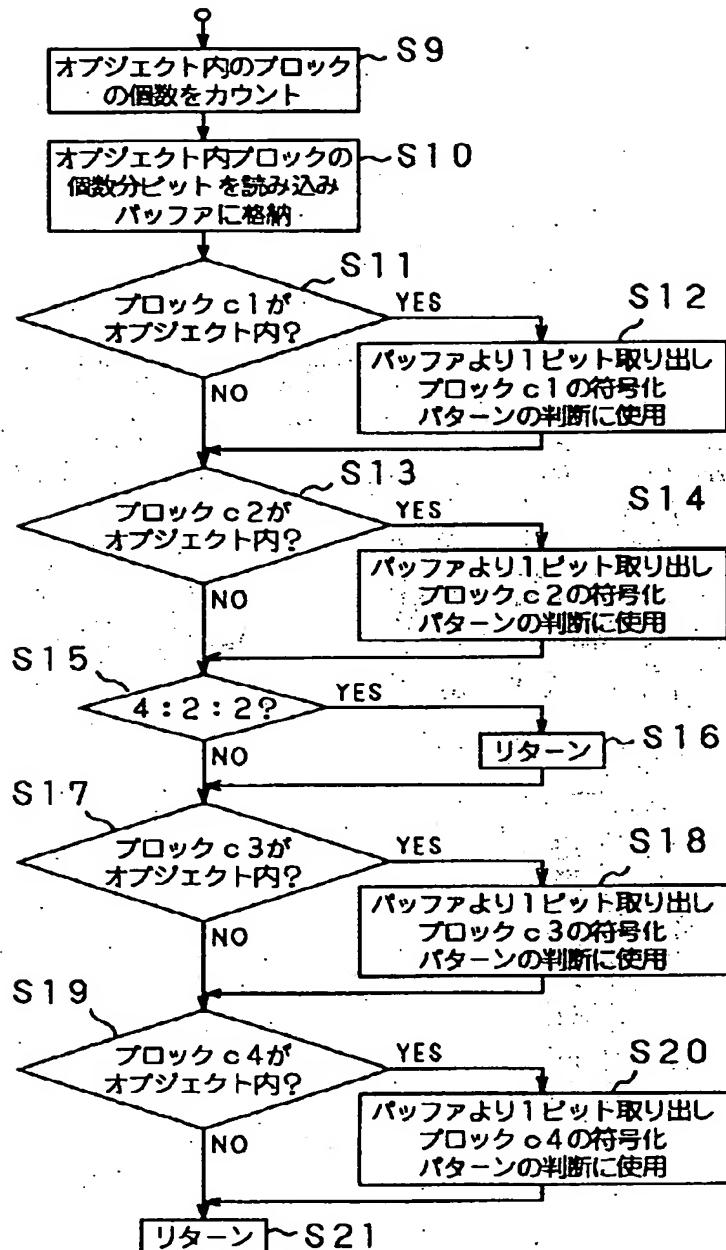
[図31]

}		
if(derived_mb_type==2) {		
for(j=0;j<4;j++)		
if(transparent_block(j))		
motion_vector("forward")		
}		
}		
for(i=0;i<block_count;i++)		
block(i)		
}		
}		
else if(co_located_not_coded!=1&(ref_select_code=='11'		
enhancement_type==1)&& scalability)) {		
if(video_object_layer_shape=="rectangular")		
mb_binary_shape_coding()		
if(video_object_layer_shape=="binary only") {		
if(transparent_mb()) {		
modb	1-2	vlclbf
if(modb) {		
if(modb>0)		
mb_type	1-4	vlclbf
if(modb==2)		
cbpb	3-6	uimsbf
if(cbpc=='10' cbpc=='11')		*
if(chroma_format=='10')		*
cbpcb	1-2	vlclbf
if(chroma_format=='11')		*
cbpcb	1-4	vlclbf
if(cbpc=='10' cbpc=='11')		*
if(chroma_format=='10')		*
cbper	1-2	vlclbf
if(chroma_format=='11')		*
cbper	1-2	vlclbf
if(ref_select_code=='00' scalability) {		*
if(mb_type=='1' && cbpb!=0)		
dquant	2	uimsbf
if(field_prediction)		
interlaced_information()		
if(mb_type=='01'		
mb_type=='0001') {		
motion_vector("forward")		

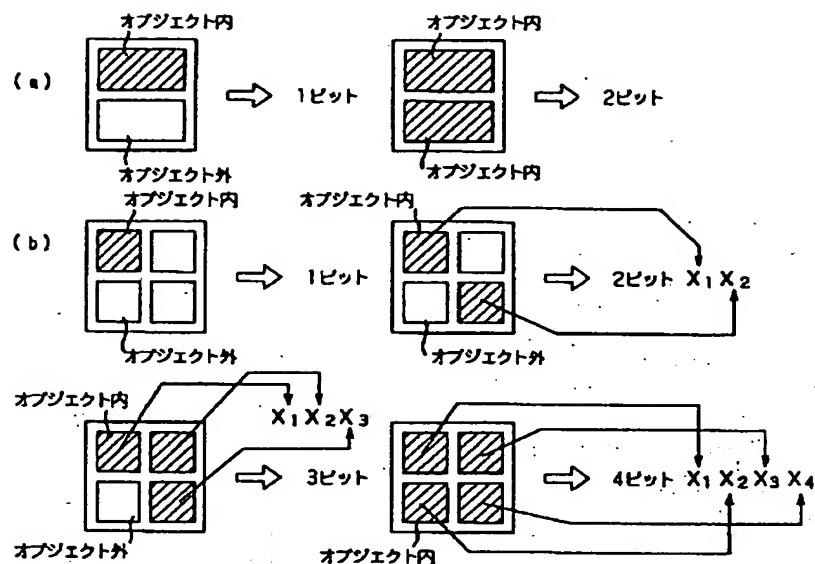
【図35】



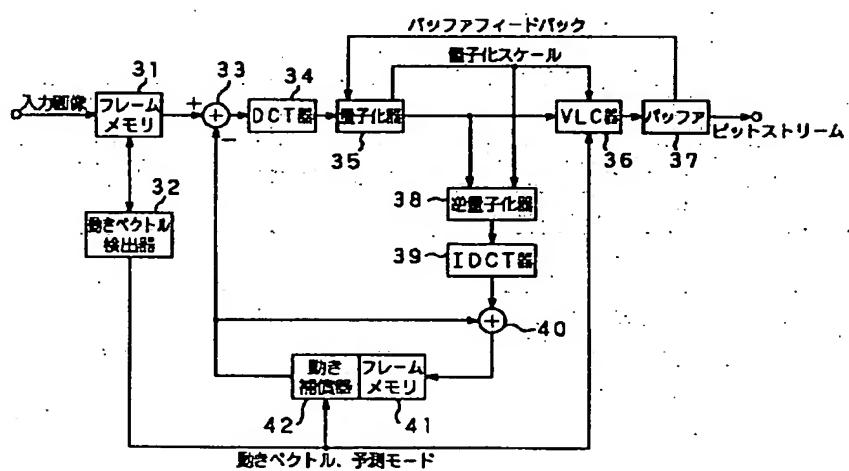
【図36】



【図38】



【図42】



【図40】

Macroblock

macroblock()	No.of bits	Mnemonic
if(VOP_coding_type == "B") {		
if(video_object_layer_shape != "rectangular" &&!(sprite_enable && low_latency_sprite_enable && sprite_transmit_mode == "update"))		
mb_binary_shape_coding()		
if(video_object_layer_shape == "binary only") {		
if(!transparent_mb()) {		
if (VOP_coding_type == "T" &&!(sprite_enable && sprite_transmit_mode == "piece"))		
not_coded	1	bsibf
if(!not_coded && VOP_coding_type == "T") {		
mcbpc	1-9	vlclbf
if(cbpc == "10" cbpc == "11")		*
if(chroma_format == "10")		*
cbpbc	1-2	vlcbf
if(chroma_format == "11")		*
cbpbc	1-4	vlcbf
if(cbpc == "01" cbpc == "11")		*
if(chroma_format == "10")		*
cbper	1-2	vlcbf
if(chroma_format == "11")		*
cbper	1-4	vlcbf
if((!short_video_header && derived_mb_type == 3 derived_mb_type == 4))		
dc_pred_flag	1	bsibf
if(derived_mb_type == "stuffing")		
cbpy	1-6	vlclbf
else		
return()		
if(derived_mb_type == 1 derived_mb_type == 4)		
dquant	2	ulmsbf
if(interlaced)		
interlaced_information()		
if((ref_select_code == "11" && scalability) && VOP_coding_type == "S") {		
if(derived_mb_type == 0 derived_mb_type == 1)		
motion_vector("forward"))		
if(field_prediction)		

【図4-1】

motion_vector("forward")		
}		
if(derived_mb_type==2) {		
for(j=0;j<4;j++)		
if(!transparent_block(j))		
motion_vector("forward")		
}		
}		
for(i=0;i<block_count;i++)		
block(i)		
}		
}		
}		
else if(co_located_not_coded!=1 (ref_select_code=='111' && enhancement_type==1) && scalability)) {		
if(video_object_layer_shape!="rectangular")		
mb_binary_shape_coding()		
if(video_object_layer_shape!="binary only") {		
if(!transparent_mb()) {		
modb	1-2	vc1bf
if(modb) {		
if(modb>0)		
mb_type	1-4	vc1bf
if(modb==2)		
cbpb	3-12	uimsbf
if(ref_select_code!='00' !scalability) {		
if(mb_type!="1" && cbpb==0)		
dquant	2	uimsbf
if(field_prediction)		
interlaced_information()		
if(mb_type=='01'		
mb_type=='0001') {		
motion_vector("forward")		
if(interlaced)		
motion_vector("forward")		
}		
if(mb_type=='01' mb_type=='001') {		
motion_vector("backward")		
if(field_prediction)		
motion_vector("backward")		
}		
if(mb_type=="1")		

プロントページの続き

(72)発明者 名雲 武文

東京都品川区北品川6丁目7番35号 ソニ

一株式会社内

Fターム(参考) 5C057 AA06 BA01 CA01 CE10 EA02
EA07 ED07 ED09 EG06 EG08
EL01 EM04 EM09 EM11 EM13
EM16 GG01
5C059 MA00 MA05 MA23 MA31 MB01
MB11 MB12 MB21 MB27 MC11
MC38 ME01 ME02 NN01 NN28
PP05 PP06 PP07 PP16 SS01
SS07 SS11 TA62 TB07 TB18
TC01 TC02 TC24 TC27 TC42
TD05 UA02 UA05 UA33
5J064 AA02 BA09 BA13 BA15 BA16
BB03 BB11 BC02 BC08 BC14
BC16 BC25 BD03

【公報種別】特許法第17条の2の規定による補正の掲載
 【部門区分】第7部門第3区分
 【発行日】平成13年7月19日(2001.7.19)

【公開番号】特開2000-23190(P2000-23190A)

【公開日】平成12年1月21日(2000.1.21)

【年通号数】公開特許公報12-232

【出願番号】特願平10-196769

【国際特許分類第7版】

HO4N 11/04

7/32

// HO3M 7/36

【F1】

HO4N 11/04 8

HO3M 7/36

HO4N 7/137 Z

【手続補正書】

【提出日】平成12年7月17日(2000.7.17)

7)

【手続補正1】

【補正対象書類名】明細書

【補正対象項目名】特許請求の範囲

【補正方法】変更

【補正内容】

【特許請求の範囲】

【請求項1】 画像の輝度成分と色差成分とをそれぞれプロック化して各プロック毎の動きベクトルを生成する動きベクトル生成方法において、

上記輝度成分のプロックに対して得られた動きベクトルに基づいて、対応する色差成分のプロックの動きベクトルを生成することを特徴とする動きベクトル生成方法。

【請求項2】 上記色差成分として、複数種類の色差フォーマットのいずれかが選択された色差成分を用い、この色差フォーマットの種類に応じて、上記輝度成分のプロックに対して得られた動きベクトルに基づく色差成分のプロックの動きベクトルの計算を行うことを特徴とする請求項1記載の動きベクトル生成方法。

【請求項3】 画像の輝度成分と色差成分とをそれぞれプロック化して各プロック毎の動きベクトルを求め、動き補償を伴う予測符号化を行う画像符号化装置において、

上記色差成分として、複数種類の色差フォーマットのいずれかが選択された色差成分を用い、

上記輝度成分のプロックの動きベクトルを検出すると共に、得られた輝度成分の動きベクトルに基づき、上記色差成分の色差フォーマットの種類に応じて選択された計算により、対応する色差成分のプロックの動きベクトルを生成する動きベクトル検出手段と、

この動きベクトル検出手段により得られた動きベクトル

を用いて動き補償を行う動き補償手段と、この動き補償手段からの出力と入力画像とに基づいて予測符号化を行う符号化部とを有することを特徴とする画像符号化装置。

【請求項4】 上記複数種類の色差フォーマットは、4:2:0フォーマット、4:2:2フォーマット、4:4:4フォーマットであることを特徴とする請求項3記載の画像符号化装置。

【請求項5】 上記動きベクトルの計算を、上記色差フォーマットの種類と、マクロプロックの予測モードとに応じて切換選択することを特徴とする請求項3記載の画像符号化装置。

【請求項6】 画像の上記輝度成分及び色差成分と共に、画像の形状情報が供給され、この形状情報に応じて、上記色差成分のプロックの動きベクトルを計算することを特徴とする請求項3記載の画像符号化装置。

【請求項7】 画像の輝度成分と色差成分とをそれぞれプロック化して各プロック毎に求められた動きベクトルに基づいて動き補償を行う動き補償方法において、上記色差成分のプロックの動きベクトルは、上記輝度成分のプロックに対して検出された動きベクトルに基づいて生成されたものであり、上記色差成分のプロックの動きベクトルに応じて、当該色差成分のプロックの動き補償を行うことを特徴とする動き補償方法。

【請求項8】 上記色差成分として、複数種類の色差フォーマットのいずれかが選択された色差成分を用い、この色差フォーマットの種類に応じて、上記色差成分のプロックの動き補償の計算方法が選択されることを特徴とする請求項7記載の動き補償方法。

【請求項9】 画像の輝度成分と色差成分とをそれぞれプロック化して各プロック毎に求められた動きベクトル

に基づいて動き補償を行う動き補償装置において、上記色差成分として、複数種類の色差フォーマットのいずれかが選択された色差成分を用い、上記色差成分のプロックの動きベクトルは、上記輝度成分のプロックに対して検出された動きベクトルに基づいて生成されたものであり、上記色差成分のプロックの動きベクトルに応じて当該色差成分のプロックの動き補償を行う際に、上記複数種類の色差フォーマットに応じた動き補償の計算方法を選択することを特徴とする動き補償装置。

【請求項10】 画像の輝度成分と色差成分とをそれぞれプロック化して各プロック毎の動きベクトルを求め、動き補償を伴う予測符号化を行うことにより得られた符号化データが記録された記録媒体において、

上記符号化データは、

上記色差成分として、複数種類の色差フォーマットのいずれかが選択された色差成分を用い、上記輝度成分のプロックの動きベクトルを検出すると共に、得られた輝度成分の動きベクトルに基づき、上記色差成分の色差フォーマットの種類に応じて選択された計算により、対応する色差成分のプロックの動きベクトルを生成し、

この動きベクトル検出手段により得られた動きベクトルを用いて動き補償を行い、

この動き補償手段からの出力と入力画像とに基づいて予測符号化を行うことにより得られたものであることを特徴とする符号化データの記録媒体。

【手続補正2】

【補正対象書類名】明細書

【補正対象項目名】0001

【補正方法】変更

【補正内容】

【0001】

【発明の属する技術分野】 本発明は、動きベクトル生成方法、画像符号化装置、動き補償方法、動き補償装置、及び記録媒体に関し、特に、例えば動画像データを光磁気ディスクや磁気テープなどの記録媒体に記録し、これを再生してディスプレイなどに表示したり、テレビ会議システム、テレビ電話システム、放送用機器、マルチメディアデータベース検索システムなどのように、動画像データを伝送路を介して送信側から受信側に伝送し、受信側において、受信された動画像データを表示したり、編集して記録するための画像符号化等に適用可能な動きベクトル生成方法、画像符号化装置、動き補償方法、動き補償装置、及び記録媒体に関する。

【手続補正3】

【補正対象書類名】明細書

【補正対象項目名】0043

【補正方法】変更

【補正内容】

【0043】 そこで、本発明はこのような状況に進みてなされたものであり、形状情報を所有している画像においての4:2:2フォーマット、4:4:4フォーマットの色差成分の符号化、及び、MPEG4において4:2:2フォーマット、4:4:4フォーマットが使用できるように拡張された画像符号化等に適用可能であり、特に、色差プロックの動きベクトル検出や動き補償に用いて好適な動きベクトル生成方法、画像符号化装置、動き補償方法、動き補償装置、及び記録媒体を提供することを目的とする。

【手続補正4】

【補正対象書類名】明細書

【補正対象項目名】0044

【補正方法】変更

【補正内容】

【0044】

【課題を解決するための手段】 本発明に係る動きベクトル生成方法及び装置は、画像の輝度成分と色差成分とをそれぞれプロック化して各プロック毎の動きベクトルを生成する際に、上記輝度成分のプロックに対して得られた動きベクトルに基づいて、対応する色差成分のプロックの動きベクトルを生成することにより、上述の課題を解決する。

【手続補正5】

【補正対象書類名】明細書

【補正対象項目名】0045

【補正方法】変更

【補正内容】

【0045】 ここで、上記色差成分として、複数種類の色差フォーマットのいずれかが選択された色差成分を用い、この色差フォーマットの種類に応じて、上記輝度成分のプロックに対して得られた動きベクトルに基づく色差成分のプロックの動きベクトルの計算を行うことが挙げられる。

【手続補正6】

【補正対象書類名】明細書

【補正対象項目名】0046

【補正方法】変更

【補正内容】

【0046】 具体的には、VOLレイヤもしくはさらに上位のレイヤにおいて、色差フォーマットを示すフラグ(croma_type)を採用する。このフラグ(croma_type)は、2ビットのフラグであり、例えばその値を表1と定義することによって符号化/復号化を行う画像の色差フォーマットを特定が可能となる。また、マクロプロックレイヤにおいても、従来、MPEG4では色差のプロックの符号化状態を表すフラグ(cbpc)を用いてその色差プロックにおけるDCT係数の存在を判断している。MPEG4では、このフラグ(cbpc)をI-VOP、P-VOPのマクロプロックの場合には、VLCである(mc

tpc) より得て、B-VOPの場合には、CBPBの下位2ビットから得ることができる。

【手続補正7】

【補正対象書類名】明細書

【補正対象項目名】0315

【補正方法】変更

【補正内容】

【0315】なお、本発明の記録媒体としては、上述した符号化データが記録された記録媒体のみならず、本発明実施の形態のエンコーダやデコーダの動作を行わせるためのプログラムが記録された記録媒体を提供することも可能である。

**This Page is Inserted by IFW Indexing and Scanning
Operations and is not part of the Official Record.**

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- BLACK BORDERS**
- IMAGE CUT OFF AT TOP, BOTTOM OR SIDES**
- FADED TEXT OR DRAWING**
- BLURRED OR ILLEGIBLE TEXT OR DRAWING**
- SKEWED/SLANTED IMAGES**
- COLOR OR BLACK AND WHITE PHOTOGRAPHS**
- GRAY SCALE DOCUMENTS**
- LINES OR MARKS ON ORIGINAL DOCUMENT**
- REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY**
- OTHER: _____**

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.